

mi COMPUTER²

CURSO PRACTICO DEL ORDENADOR PERSONAL,
EL MICRO Y EL MINIORDENADOR



150 ptas

Editorial  Delta, s.a.

mi COMPUTER

CURSO PRACTICO DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Publicado por Editorial Delta, S.A., Barcelona, y comercializado en exclusiva por Distribuidora Olimpia, S.A., Barcelona

Volumen I - Fascículo 2

Director: José Mas Godayol
Director editorial: Gerardo Romero
Jefe de redacción: Pablo Parra
Coordinación editorial: Jaime Mardones
Asesor técnico: Roberto Quiroga

Redactores y colaboradores: G. Jefferson, R. Ford, S. Tarditti, A. Cuevas

Para la edición inglesa: R. Pawson (editor), D. Tebbutt (consultant editor), C. Cooper (executive editor), D. Whelan (art editor), Bunch Partworks Ltd. (proyecto y realización)

Realización gráfica: Luis F. Balaguer

Redacción y administración:
Paseo de Gracia, 88, 5.^o - Barcelona-8
Tels. (93) 215 10 32 / (93) 215 10 50 - Télex 97848 EDLTE

MI COMPUTER, *Curso práctico del ordenador personal, el micro y el miniordenador*, se publica en forma de 96 fascículos de aparición semanal, encuadrables en ocho volúmenes. Cada fascículo consta de 20 páginas interiores y sus correspondientes cubiertas. Con el fascículo que completa cada uno de los volúmenes, se ponen a la venta las tapas para su encuadernación.

El editor se reserva el derecho de modificar el precio de venta del fascículo en el transcurso de la obra, si las circunstancias del mercado así lo exigieran.

© 1983 Orbis Publishing Ltd., London
© 1984 Editorial Delta, S.A., Barcelona
ISBN: 84-85822-83-8 (fascículo) 84-85822-84-6 (tomo 1)
84-85822-82-X (obra completa)
Depósito Legal: B. 52-84

Fotocomposición: Tecfa, S.A., Pedro IV, 160, Barcelona-5
Impresión: Cayfosa, Santa Perpètua de Mogoda (Barcelona) 238401
Impreso en España - Printed in Spain - Enero 1984

Editorial Delta, S.A., garantiza la publicación de todos los fascículos que componen esta obra.

Distribuye para España: Marco Ibérica, Distribución de Ediciones, S.A., Carretera de Irún, km 13,350. Variante de Fuencarral, Madrid-34.

Distribuye para Argentina: Viscontea Distribuidora, S.C.A., La Rioja 1134/56, Buenos Aires.

Distribuye para Colombia: Distribuidoras Unidas, Ltda., Transversal 93, n.^o 52-03, Bogotá D.E.

Distribuye para México: Distribuidora Intermex, S.A., Lucio Blanco, n.^o 435, Col. San Juan Tlhuaca, Azcapotzalco, 02400, México D.F.

Distribuye para Venezuela: Distribuidora Continental, S.A., Ferrenquín a Cruz de Candelaria, 178, Caracas, y todas sus sucursales en el interior del país.

Pida a su proveedor habitual que le reserve un ejemplar de MI COMPUTER. Comprando su fascículo todas las semanas y en el mismo quiosco o librería, Vd. conseguirá un servicio más rápido, pues nos permite realizar la distribución a los puntos de venta con la mayor precisión.

Servicio de suscripciones y atrasados (sólo para España)

Las condiciones de suscripción a la obra completa (96 fascículos más las tapas, guardas y transferibles para la confección de los 8 volúmenes) son las siguientes:

- a) Un pago único anticipado de 16 690 ptas. o bien 8 pagos trimestrales anticipados y consecutivos de 2 087 ptas. (sin gastos de envío).
- b) Los pagos pueden hacerse efectivos mediante ingreso en la cuenta 3371872 de la Caja Postal de Ahorros y remitiendo a continuación el resguardo o su fotocopia a Distribuidora Olimpia (Paseo de Gracia, 88, 5.^o, Barcelona-8), o también con talón bancario remitido a la misma dirección.
- c) Se realizará un envío cada 12 semanas, compuesto de 12 fascículos y las tapas para encuadrarlos.

Los fascículos atrasados pueden adquirirse en el quiosco o librería habitual, sin variación alguna sobre el precio de venta en vigor en el momento de la petición. También pueden recibirse por correo, con incremento del coste de envío, haciendo llegar su importe a Distribuidora Olimpia, en la forma establecida en el apartado b). Para cualquier aclaración, telefonear al (93) 215 75 21.

No se efectúan envíos contra reembolso.

REM es la abreviatura de REMark (observación). Todo lo que aparece en la misma línea después de REM es ignorado por el ordenador. Las observaciones constituyen una manera muy práctica de hacernos recordar lo que está haciendo el ordenador. Esta REM en particular no es más que un título, no nos indica lo que está haciendo el programa. Ya avanzado el curso, veremos la gran ayuda que representan unas REM bien escritas. Ahora echemos una mirada a:

20 PRINT "DIGITE UN NUMERO"

Cuando el BASIC llegue a la palabra PRINT, todo lo que sigue a continuación es "impreso" en la pantalla del ordenador. Observe que la oración está entre comillas dobles. Una de las reglas del BASIC es que los caracteres (letras) que aparecen entre comillas dobles después de la enunciación PRINT aparecerán en pantalla exactamente como fueron digitados. En la línea 60 veremos otra forma de utilizar PRINT. A continuación sigue:

30 INPUT A

Por ahora pasaremos esta línea por alto; volveremos a ella después de ocuparnos de la línea 40.

40 LET A = A + 1

Aquí la letra A está utilizada como una variable. Una variable es como una caja etiquetada que puede tener tanto un número como algunos caracteres. En lugar de tener que recordar qué contiene la caja, lo que tenemos que saber es cómo se llama la caja para poder referirnos a ella. Es algo así como decir "Alcántame la caja que lleva la etiqueta B" en lugar de decir "Alcántame la caja que contiene los tornillos de 15 mm de cabeza".

En esta línea tenemos una «caja» denominada A. Esta caja se llama variable porque el valor de lo que pongamos en ella puede variar. A una variable podemos asignarle virtualmente cualquier valor. En la línea 30 le habíamos asignado un valor a la variable A, así que volvamos a ella:

30 INPUT A

La utilización de la palabra INPUT es, en BASIC, uno de los modos de asignarle (otorgarle) a una variable un valor específico. Cuando el programa BASIC llega a una línea que comienza con INPUT, espera a que algo sea digitado en el teclado. INPUT A permite que el ordenador sepa que tenemos una variable denominada A y que lo que se digite en el teclado será el valor asignado a esa variable. En este punto, digitando 7<CR> coloca 7 en la caja A o, utilizando términos informáticos, asigna el valor 7 a la variable A. Ahora que ya sabemos en qué consiste una variable y que conocemos uno de los modos de asignarle un valor, volvamos a ocuparnos de la línea 40.

40 LET A = A + 1

El nombre de la variable a la cual se le asigna un valor aparece siempre a la izquierda del signo igual. Aquí le estamos dando a A un nuevo valor. La sentencia significa "Hagamos (LET) que el nuevo valor de A sea igual al antiguo valor más 1". El antiguo valor de A era 7. Ahora lo hemos convertido en 7 + 1, de modo que el nuevo valor es 8.

50 PRINT "CREO QUE EL NUMERO DIGITADO ERA";

Ésta es, nuevamente, nuestra sentencia de impresión. Esta sentencia "imprime" la secuencia de caracteres

(es decir, las palabras o los números que usted ha digitado) entre las comillas dobles. Observemos el punto y coma al final de la línea. Éste ayuda a especificar las posiciones en las cuales las cosas se imprimen en pantalla. Ya avanzados en el curso, retomaremos con mayor detalle el uso del punto y coma. Ahora ocupémonos de:

60 PRINT A

Aquí hay otra sentencia PRINT, pero esta vez A no está entre comillas. Ya sabemos que el programa no imprimirá en pantalla una A verdadera, porque hemos visto ya que para ello se requiere el entrecomillado. Sin las comillas, el BASIC busca una variable que posea la misma etiqueta que el carácter que sigue a PRINT. Si lo encuentra, imprime el valor de la variable. (¡De no encontrarlo, proporcionaría un mensaje de error!) Este programa ya posee una variable denominada A y por ello el BASIC imprime su valor. ¿Y cuál es ese valor?

Si usted cree que la respuesta es 7, recuerde que el BASIC trabaja con los programas línea a línea, siguiendo el orden de los números de línea. Llegados a la

El recuadro inferior muestra cómo se utilizan las "variables" en BASIC. También ilustra cómo se utiliza la sentencia GOTO (véase página siguiente) para formar un bucle

Variables

20 PRINT
"DIGITE UN
NUMERO"

30 INPUT A

40 LET A = A + 1

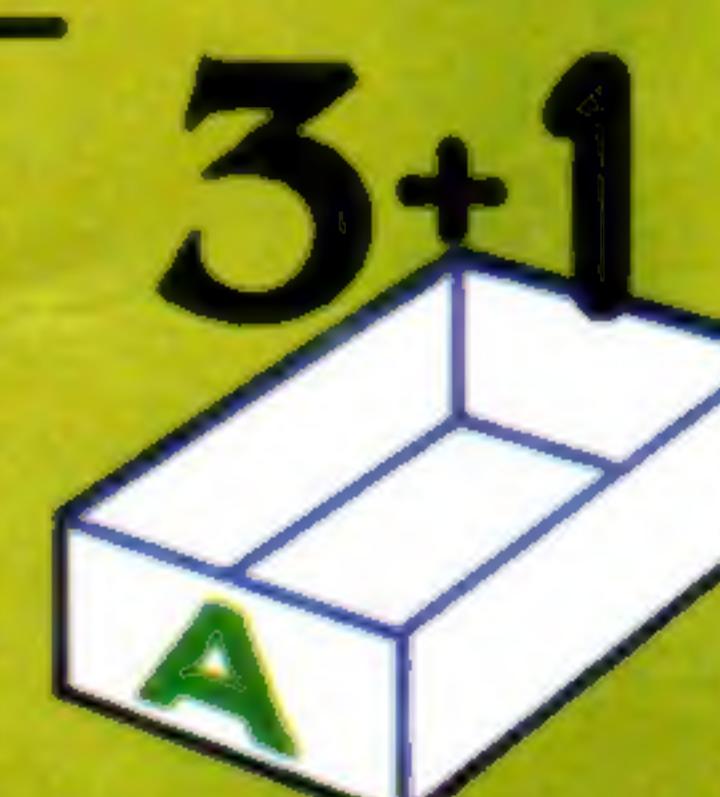
50 PRINT "CREO QUE EL..."
60 PRINT A
70 GOTO 20



Esta línea imprime el mensaje entre comillas



Cuando se ha digitado un número y se ha pulsado RETURN, el número es asignado a la variable



Esta sentencia significa "hagamos que el nuevo valor de A sea igual al antiguo valor de A más 1". Como el número pulsado era 3, el nuevo valor de la variable A es 4



50 imprime el mensaje
60 imprime el valor de la variable
70 hace un bucle hasta la línea 20, es decir, comienza de nuevo

Complementos al BASIC

LET

Sólo el Sinclair Spectrum utiliza el elemento LET de la instrucción. En otros ordenadores, éste está implícito, es decir, puede pasarse por alto. Por ejemplo, la línea 40 se podría escribir A = A + 1 en lugar de LET A = A + 1

END

Este elemento no se utiliza en el Spectrum. Se supone que la última línea digitada del programa es el final de éste

GOTO

En el Spectrum aparece en la pantalla como dos palabras separadas (GO TO), aunque sólo se oprime una tecla. La mayoría de los otros ordenadores aceptan que la instrucción se digite en dos palabras

línea 60, el valor de A ya había sido cambiado a 8, y eso es lo que el ordenador imprimirá. Finalmente llegamos a:

70 END

La sentencia END (fin) le dice al BASIC que se ha llegado al final del programa. Algunas versiones de BASIC insisten en que todos los programas deben terminar en END, mientras que otras no (ver el recuadro "Complementos al BASIC").

Observe que cuando usted pone en funcionamiento el programa, éste sólo lo hace una vez. Para que funcione otra vez, debe volver a digitar RUN<CR>.

Veamos ahora una manera de hacer funcionar el programa todas las veces que deseemos utilizando la sentencia GOTO.

Utilizando GOTO

A continuación proporcionamos el mismo programa pero con una línea adicional. Si usted había apagado el ordenador para tomar un descanso, digítelo. De no ser así, bastará con teclear las líneas 70 y 80. Éstas aparecen en azul en el siguiente listado.

```
10 REM LOS ORDENADORES NUNCA SE
EQUIVOCAN<CR>
20 PRINT "DIGITE UN NUMERO"<CR>
30 INPUT A<CR>
40 LET A = A + 1<CR>
50 PRINT "CREO QUE EL NUMERO
DIGITADO ERA";<CR>
60 PRINT A<CR>
70 GOTO 20<CR>
80 END<CR>
```

Después de haberlo digitado y listado (LIST), trate de imaginarse lo que sucederá antes de intentar llevarlo (RUN). Luego pulse RUN<CR> y, al igual que en la primera versión del programa, aparecerá:

DIGITE UN NUMERO

Digite cualquier número (utilizando las teclas numéricas) y oprima RETURN. El ordenador le sumará 1 al número y lo expondrá al final del mensaje.

CREO QUE EL NUMERO DIGITADO ERA 8

Esto es seguido inmediatamente por un nuevo men-

saje DIGITE UN NUMERO. Digitando otro número y volviendo a apretar RETURN, el programa repetirá este ciclo *ad infinitum*. La razón para que se produzca esta repetición la hallaremos en la línea 70:

70 GOTO 20

Cuando el BASIC llega a una sentencia GOTO, en lugar de continuar con la línea siguiente, va hacia (GOes TO) la línea cuyo número se especifica. En este caso se dirige nuevamente hasta la línea 20 y hace que todo el programa sea repetido otra vez. Y así continuaría indefinidamente realizando estos bucles. Si usted desea dejar de llevar el programa, se encontrará con que no existe manera de interrumpir estos bucles. El programa sigue adelante esperando su entrada.

Como usted comprenderá, existen maneras de escribir el programa que permitan pararlo cuando lo deseemos, y de ellas nos ocuparemos en el próximo fascículo. Mientras, deberemos detener el programa. Si el ordenador utilizado tiene una tecla BREAK, puede usarla para hacer que el programa deje de funcionar. Digitando RUN<CR> el programa comenzará nuevamente.

Observe que aún tenemos la expresión END al final del programa. Tal como hemos escrito este programa, con la sentencia GOTO 20 creando un bucle interminable, nunca llegaremos hasta el final, ¡pero algunas versiones de BASIC insisten en que utilicemos siempre un END al final!

Si no encuentra una manera de detener el programa, inténtelo oprimiendo la tecla RESET. Con ello es prácticamente seguro que el programa se detendrá. Luego pulse nuevamente LIST. Si obtiene un listado, usted podrá "editar" el programa en los ejercicios que proporcionamos a continuación. Si no obtiene un listado, ello significa que el RESET de su ordenador destruye el programa en memoria y, por lo tanto, usted deberá digitarlo completo otra vez.

Ejercicios

Estos ejercicios han sido cuidadosamente seleccionados por grados y están concebidos para que resulten amenos. Intentar resolverlos es una de las mejores maneras de verificar si usted ha comprendido el material que le hemos presentado y si efectivamente avanza en sus conocimientos de informática.

Antes de comenzar con los ejercicios, trate de modificar unas pocas líneas para ver qué efectos producen estos cambios en la forma de llevar el programa. No existe ninguna posibilidad de que el ordenador resulte dañado, aunque usted cometa errores o pulse teclas equivocadas. Para cambiar una línea, digite el programa y luego verifique el resultado haciendo el listado (LIST). En la pantalla volverá a aparecer todo el programa. Teclee el número de la línea que desea cambiar, seguido de la nueva línea. Intente con esta sentencia:

10 REM LOS ORDENADORES ALGUNAS VECES SE
EQUIVOCAN<CR>

luego digite LIST otra vez. Observe cómo ha cambiado la primera línea. Si desea borrar toda la línea, basta con que pulse el número de línea seguido de <CR>. Pruebe:

10<CR>
LIST

La línea 40 debería desaparecer. Ponga de nuevo la línea 10 volviendo a digitar toda la línea al completo; ¡y no se olvide del número de línea!

■ Reescriba el programa de modo que el ordenador imprima realmente el número digitado. Una pista: el truco está en quitar una línea entera.

■ Vuelva a digitar la línea 70, de modo que el programa pase a la línea 80. Haga el listado del programa (LIST). Haga funcionar el programa (RUN). ¿Por qué el programa no se ha comportado de la misma manera que antes?

■ Modifique la línea 60 para que el ordenador imprima en pantalla una A en lugar del valor de la variable A.

■ Reescriba la línea 60 de modo que el ordenador vuelva a imprimir el valor de la variable A. Quite por completo la línea 10 (la línea de REM). Digite RUN. ¿Hay alguna diferencia en la forma en que se lleva el programa?

■ Incluya una nueva observación (REM) en la línea 25. Pueden agregarse nuevas líneas con sólo digitar el nuevo número seguido de la nueva sentencia. Coloque una observación en la línea 25 para recordarle lo que ocurrirá luego; podría ser algo así como "espera una

entrada proveniente del teclado". Después de digitar la nueva línea y de pulsar <CR>, vuelva a listar (LIST) el programa y verifique que su nueva observación aparezca en el lugar correcto.

■ Reescriba el programa de modo que multiplique por 10 el número que usted digite. Necesitará modificar la línea 50 para imprimir algo parecido a **EL NUMERO QUE USTED DIGITO MULTIPLICADO POR 10 ES**. En esta ocasión no deseamos sumar al valor de la antigua variable, sino multiplicarlo por 10. Para designar "multiplicado por", el BASIC utiliza el signo *. (No utilice una X, porque el BASIC sólo reconoce en ella a una letra y no al signo de multiplicación).

Hemos recorrido ya un gran trecho. Hemos visto cómo escribir comentarios, que para el BASIC son observaciones (REMARKS), cómo imprimir (PRINT) secuencias de caracteres en la pantalla, cómo imprimir (PRINT) el valor de una variable en pantalla y cómo hacer que el programa vaya hacia (GOTO) un número de línea determinado.

El próximo tema del que nos ocuparemos es cómo salir de un bucle utilizando una expresión IF-THEN. Descubriremos cómo hacer que el programa "realice" para nosotros un número de ciclos determinados en lugar de seguir haciendo bucles de manera indefinida.

Y entonces surgió el BASIC

En la actualidad, el BASIC es el lenguaje de programación más popular del mundo. Los lenguajes de ordenador se inventaron para que el operador humano pudiera comunicarse más fácilmente con la máquina, y el BASIC es uno de los lenguajes más sencillos de aprender y de utilizar. Consiste en una serie de instrucciones simples en inglés combinadas, cuando es necesario, con los símbolos matemáticos del teclado de una máquina de escribir.

El BASIC es un lenguaje al que se llega a dominar rápidamente. A los pocos minutos de haber desembalado un microordenador, usted puede estar escribiendo programas sencillos. Fue desarrollado en 1965 en el Dartmouth College de New Hampshire (Estados Unidos), con el único fin de simplificar los lenguajes ya existentes. Sus inventores fueron dos maestros, Thomas Kurtz y John Kemeny. El uso universal del BASIC ha supuesto la introducción en el mismo de ligeras variaciones de lenguaje. Pero el núcleo del lenguaje BASIC sigue siendo respetado por todos los fabricantes.

Un programa es una secuencia de instrucciones que el ordenador ejecuta para llevar a cabo una tarea específica. La tarea puede ser realizar una previsión financiera mensual o trasladar a un "invasor del espacio" a través de la pantalla del televisor. El programa aparece como una serie de líneas numeradas. Las órdenes se aprenden rápidamente y hasta el programa más complicado no utiliza más que



El BASIC ha desmitificado la programación y ha hecho del ordenador algo accesible a cualquier persona

combinaciones y repeticiones de las órdenes elementales.

La mayoría de los ordenadores salen de fábrica con el BASIC incorporado. Los ordenadores también pueden programarse en "código de lenguaje máquina" (al que se describe como un lenguaje de "bajo nivel", porque su estructura se asemeja a la lógica de los circuitos electrónicos). El BASIC es un lenguaje de "alto nivel" porque estructuralmente se aproxima al inglés cotidiano. Para aplicaciones técnicas y especializadas existen muchos otros lenguajes de alto nivel, pero el BASIC es el lenguaje idóneo para poder introducir todos los demás. Es un lenguaje sencillo y poderoso.

La revolución mundial

La revolución de los ordenadores se extiende por toda la Tierra y está cambiando los hábitos de la sociedad. Es el mundo del mañana, pero se vive hoy

La revolución de los ordenadores coincidió con la carrera espacial y con el viaje que llevó al primer hombre a la Luna. Los millones de dólares que se invirtieron en este esfuerzo tuvieron como consecuencia directa la concentración de las mejores mentes científicas y la apertura de una brecha en cuanto a técnicas de fabricación. El viaje a la Luna hizo posible algo casi imposible. Los efectos inmediatos abarcan desde la creación de nuevos materiales de cerámica, plásticos y adhesivos hasta la microminiaturización de un increíble potencial informático.

Cuando un "ordenador" se identificaba con bastidores llenos de circuitos, también implicaba miles de componentes electrónicos unidos entre sí mediante cables individuales. Los costos de fabricación eran enormes. Ahora esos mismos circuitos pueden introducirse en chips de silicio lo suficientemente pequeños, como para caber en una carcasa del tamaño de un teclado.

Los chips albergados en el interior de ordenadores como el Dragon o el Spectrum no sólo son pequeños, sino que pueden fabricarse en serie a un costo casi insignificante. El silicio, la materia prima a partir de la cual se producen los chips, es uno de los elementos más comunes de la Tierra. Los granos de arena están compuestos en su mayor parte por silicio.

Ahora que el ordenador ha conquistado un sitio en los hogares y en las oficinas de todo el mundo desarrollado, tenemos la oportunidad de constituirnos en privilegiados testigos del comienzo de la segunda revolución industrial. La primera revolución industrial sustituyó la mano de obra del trabajador por una maquinaria motorizada. La revolución del ordenador supondrá un ahorro del tiempo de trabajo del personal especializado y reemplazará a los obreros de las fábricas por robots controlados por ordenador.

La incidencia a largo plazo de esta revolución en nuestras vidas no está ni mucho menos esclarecida. Lo único que puede suponerse con toda certeza es que los patrones de trabajo y ocio sufrirán modificaciones y que éstas se producirán con rapidez. Los robots, extensiones mecánicas de los ordenadores, están reemplazando a los obreros. Éstos se enfrentan a la disyuntiva de especializarse en algo o quedarse sin trabajo.

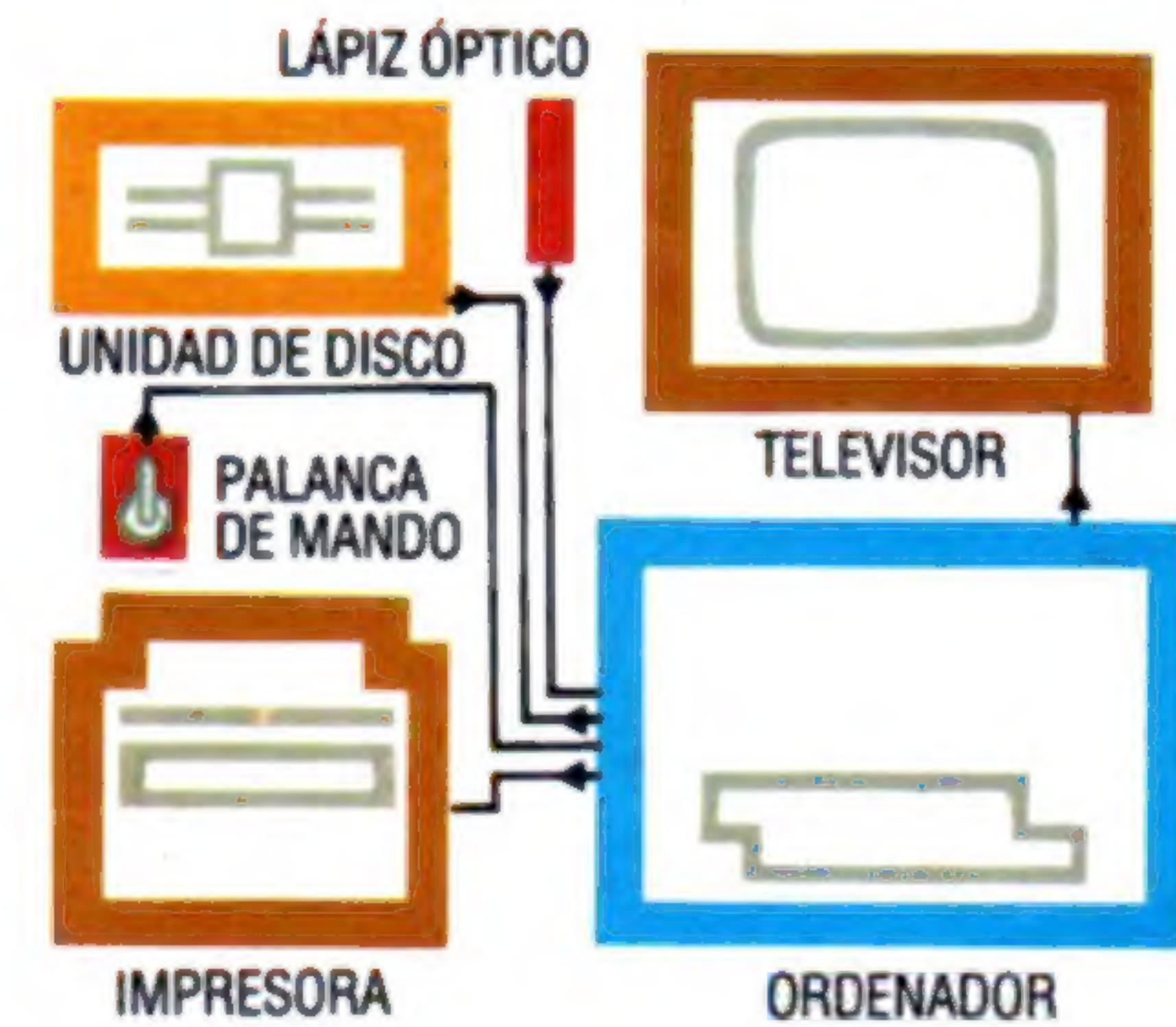
Actividades tan tradicionales como las de tipógrafo y cajista, e incluso la de maestro, podrían ser sustituidas con la actual tecnología. Y dentro de poco acudir a la consulta del médico de cabecera podrá consistir en una entrevista con un terminal de ordenador.

La consecuencia social de la primera revolución industrial fue el desplazamiento de millones de personas desde las zonas rurales y la aparición de la polución industrial (y los beneficios materiales) que caracterizan al mundo occidental. Nos hallamos al borde de una revolución tan dramática como aquélla, en la que el motor de la sociedad será el ordenador. Una revolución en la que la clave para la supervivencia estará en comprender al ordenador y saber utilizarlo.

- Ordenador
- Unidad de salida
- Unidad de entrada
- Unidad de entrada o salida
- Interface

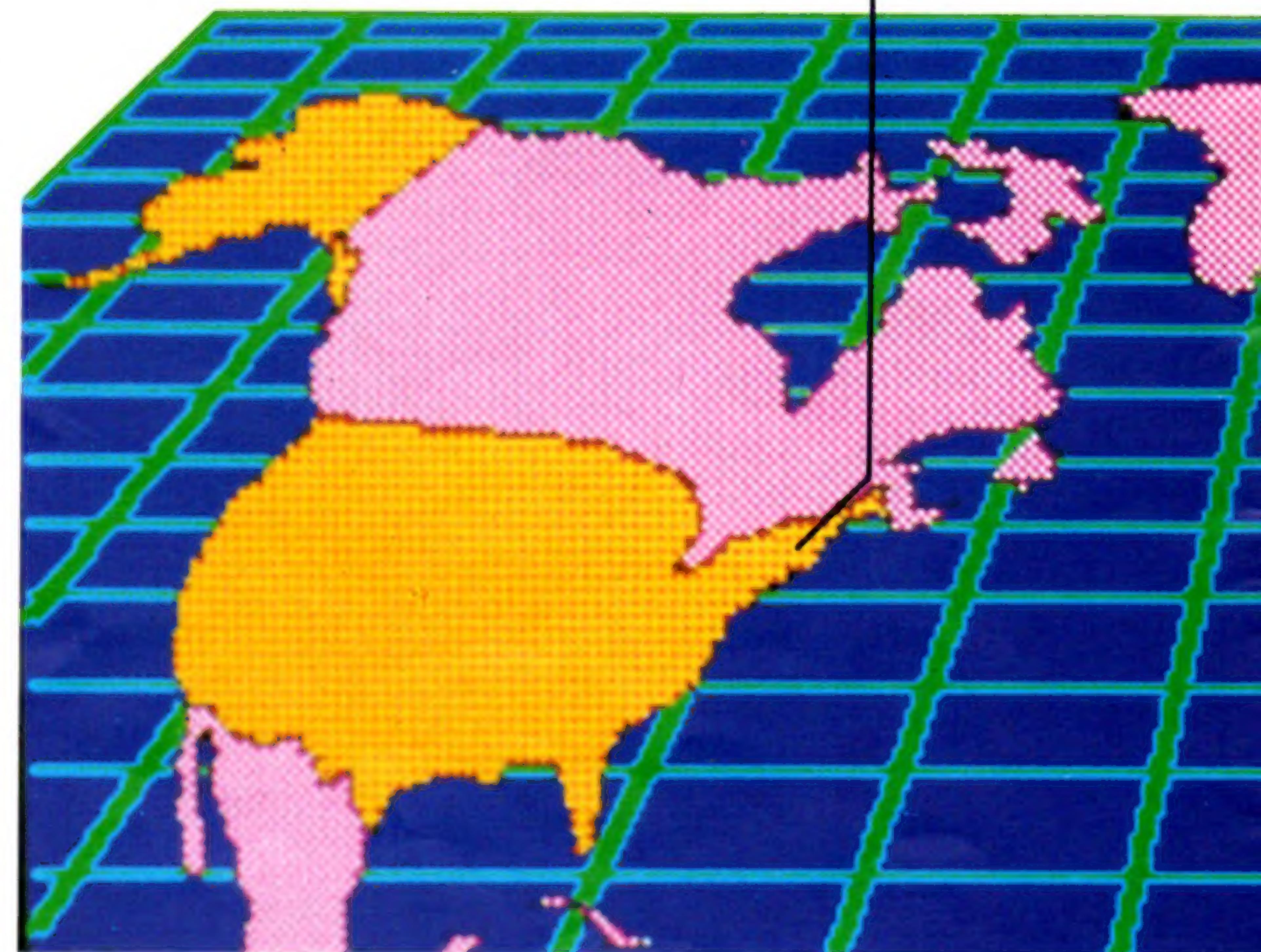
Personal

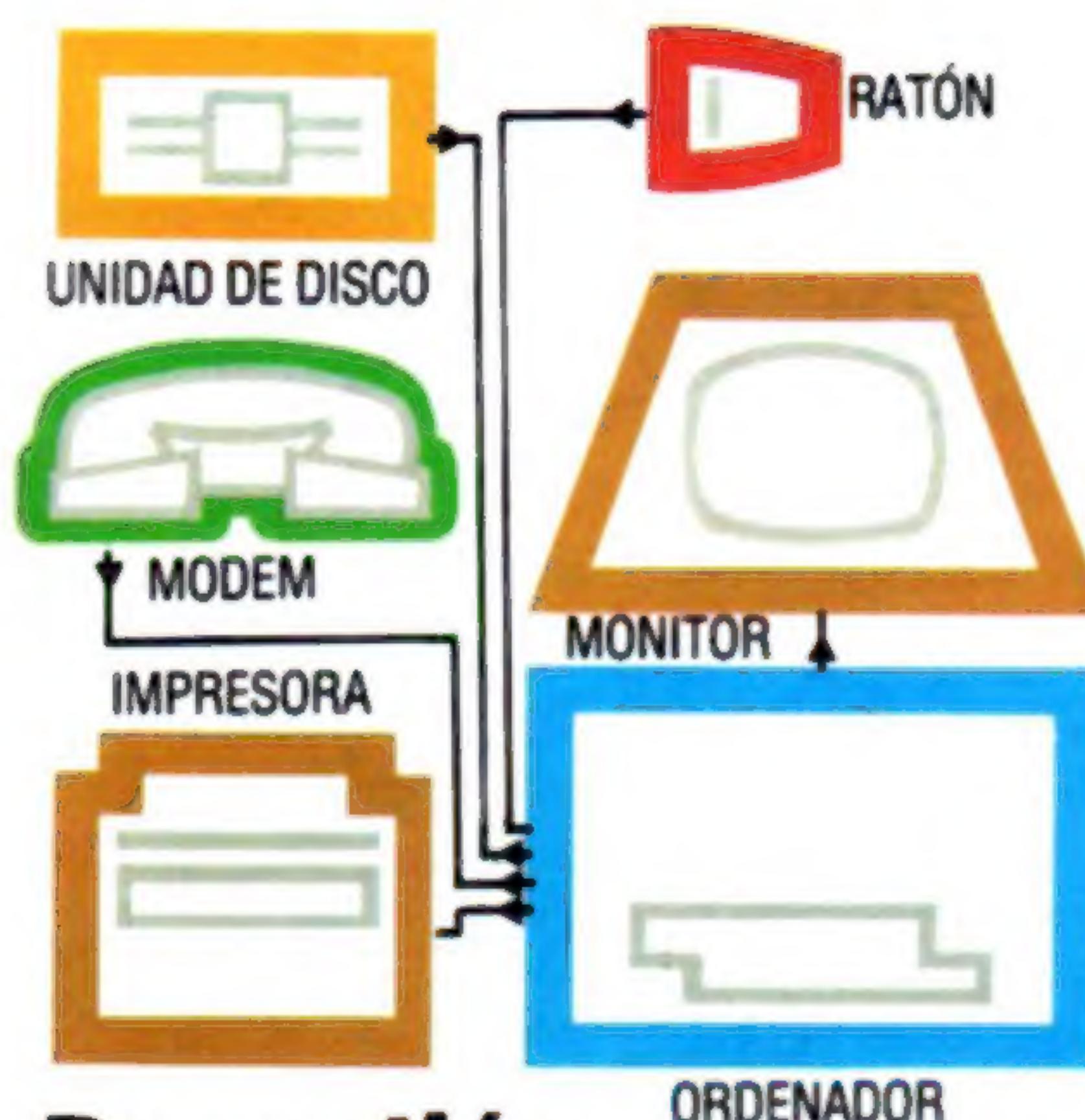
Un sistema personal económico puede montarse por poco dinero. Es su pasaporte para el futuro



Juegos

Bastan unos pocos accesorios para acceder a su mundo. Los juegos no sólo entretienen, sino que enseñan los principios básicos de la informática



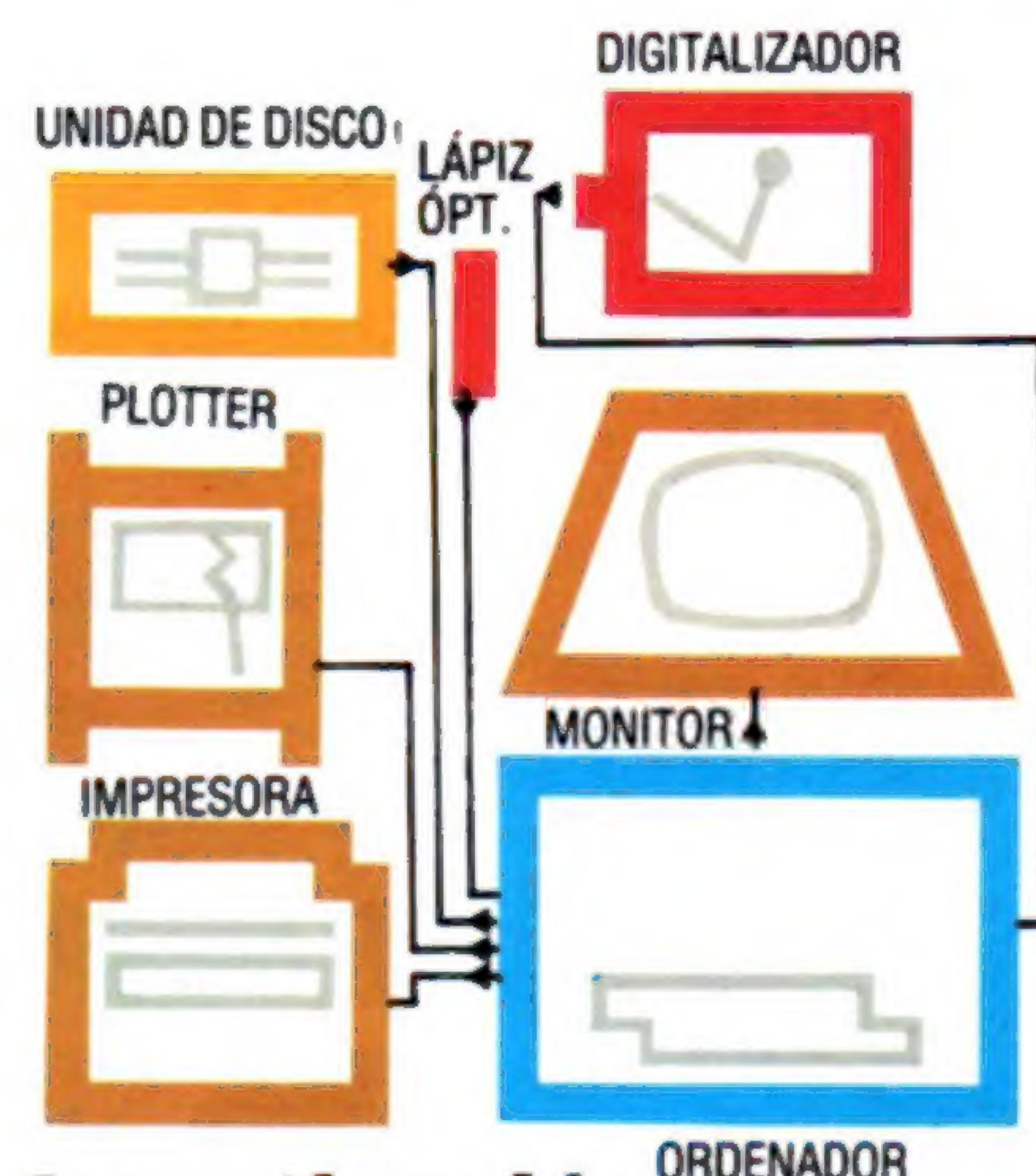
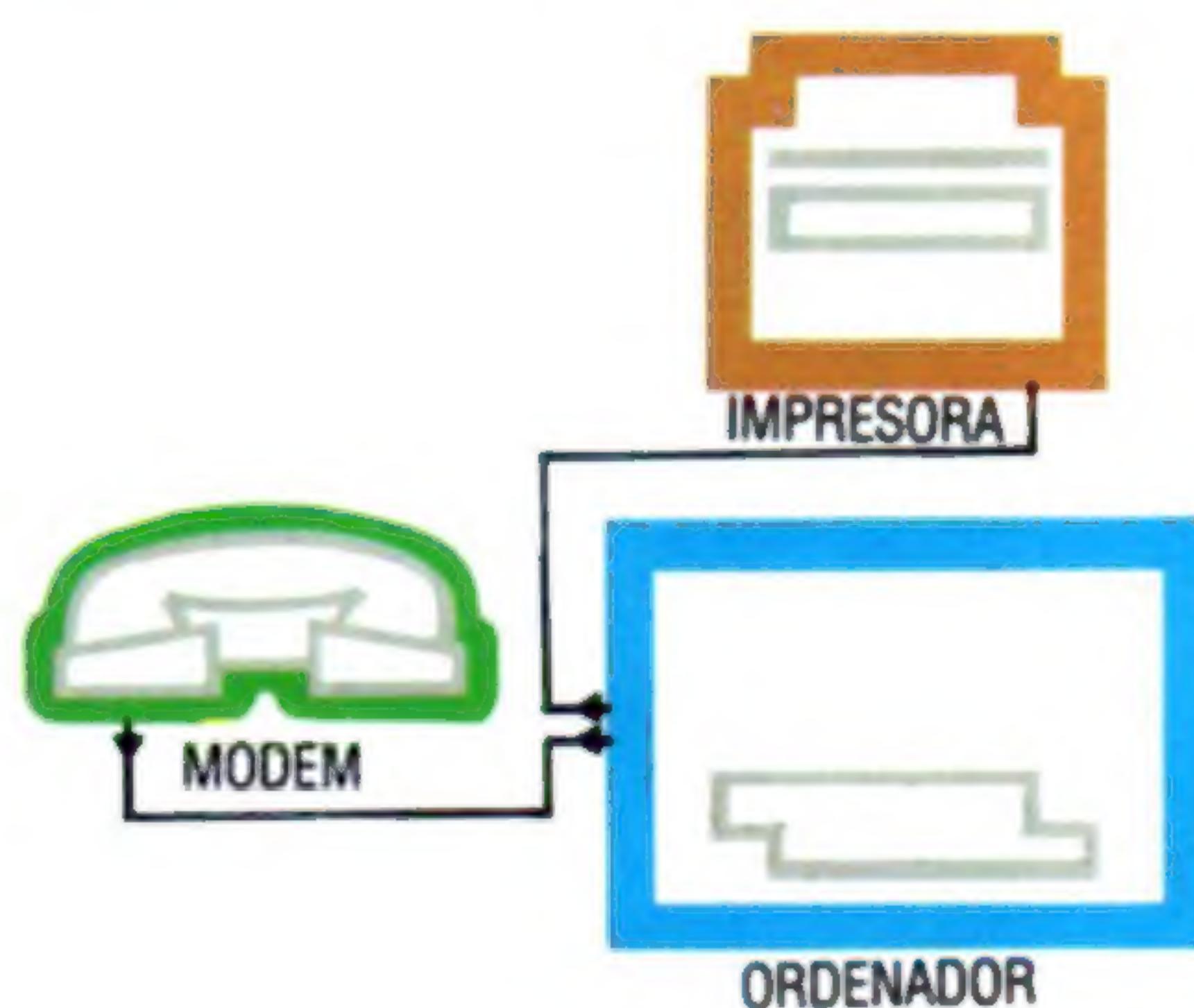


De gestión

Los pequeños sistemas de microordenador están ayudando (y reemplazando) a los empleados administrativos y a los hombres de negocios. Los modems hacen posible el acceso, a través de un teléfono convencional, a ordenadores grandes y a la imprescindible información de sus enormes bancos de datos. Los equipos de mecanógrafos se van convirtiendo en algo ya anticuado, a medida que el tratamiento de textos va ganando terreno. El software contable y por "hojas electrónicas" reduce los costos de la contabilidad y permite que las decisiones financieras se adopten en cuestión de minutos. Las unidades de disco sustituyen salas enteras repletas de archivadores, y la impresora se encarga de todo, desde imprimir cartas perfectas hasta proporcionar una tabulación instantánea de los coeficientes del día en la bolsa de valores. Y el empresario que no sepa digitar será auxiliado por el «ratón», dispositivo similar al mando de bola que permite introducir las órdenes en la máquina.

Comunicación

La red de teléfonos es internacional y ahora muchos ordenadores son tan pequeños que pueden sostenerse en la mano o en el regazo. Un atareado ejecutivo puede digitar sus informes a bordo de un Boeing 747 que lo lleva a Nueva York y, a su llegada, enviarlos a casa a la velocidad del electrón. El modem comunica el ordenador personal con el teléfono y permite en teoría un acceso instantáneo a todos los ordenadores del mundo, incluyendo a los ordenadores industriales gigantes. Los bancos de datos están a mano las 24 horas del día y hasta un periodista, que esté consultando el archivo desde Afganistán o Sudamérica, puede redactar su artículo en el tiempo que tarde en hallar una cabina telefónica.



Investigación

El potencial de un ordenador avanzado ya ha dejado de ser patrimonio exclusivo del laboratorio de una universidad o de una multinacional. En las escuelas y en los hogares de todo el mundo se están introduciendo sofisticados sistemas. Incluso es posible investigar por CAD (Computer Aided Design, diseño auxiliado por ordenador) y AI (Artificial Intelligence, inteligencia artificial) con un sistema personal. También son asequibles los "plotters" (dispositivos trazadores de gráficos), para realizar dibujos técnicos, y los digitalizadores, que permiten dar entrada a imágenes complejas, como pueden ser mapas o diagramas. En la actualidad, los conocimientos en materia de ordenadores son tan importantes para un currículum de estudios como lo fuera el latín hasta hace algunos años. Los niños del presente están aprendiendo a enfrentarse con el mundo informatizado del mañana.



2,2 m
PERSONAL

FRANCIA

0,5 m
DE GESTION

R.F. ALEMANIA

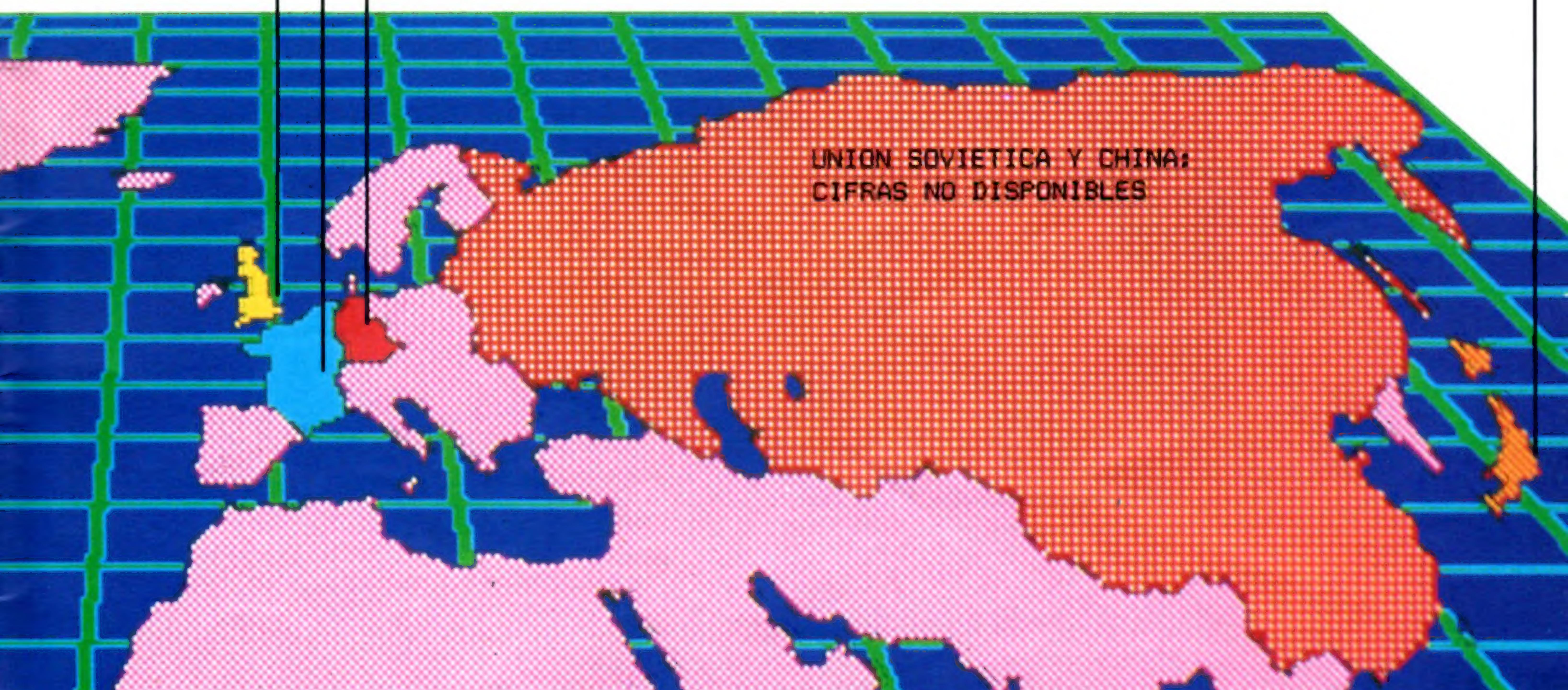
0,6 m
JUEGOS

JAPON



1,1 m
DE GESTION

UNION SOVIETICA Y CHINA:
CIFRAS NO DISPONIBLES



El artista electrónico

Sofisticadas o sencillas, las imágenes por ordenador se componen de miles de puntos diminutos. Y el ordenador debe recordar el color y el brillo de cada uno



Pintando con un micro

Pintar imágenes con un microordenador es sencillo si se dispone de un software para gráficos. Estos programas permiten dibujar complicadas imágenes a través de las órdenes que se digitán en el teclado. Por lo general se empieza colocando líneas rectas en la pantalla. Pueden agregarse círculos, triángulos, cuadrados y otras formas predefinidas. El color puede utilizarse tanto para las líneas como para zonas completas de la pantalla. Las zonas pueden pintarse seleccionando un color, ya sea de una paleta visualizada en pantalla o a través del teclado. La zona que queda encerrada entre sus líneas es coloreada de forma automática. En algunos programas se puede mover un "pincel" a través de la pantalla, como un cursor. La fotografía grande muestra el resultado de este proceso de "construcción".



Usted está pilotando un aeroplano en vuelo rasante sobre las azoteas de los edificios; se le ha detenido un motor; enfrente de usted se dibuja la silueta amenazante de un rascacielos... y en medio de la pista en la que pretendía desesperadamente aterrizar está ardiendo otro avión. Todo esto sucede ante sus ojos, en la pantalla del televisor. Es una forma de gráficos por ordenador.

En el caso de los programas educativos, como los destinados a que los niños aprendan a deletrear o a realizar operaciones aritméticas, es evidente que, a menos que un programa sea visualmente atractivo, no podrá mantener la atención de los pequeños durante mucho tiempo.

Es probable que quienes utilizan los microordenadores con fines empresariales traten básicamente con números que representen las cantidades de dinero que se han recibido o se han gastado, el stock de un artículo determinado, etc. Pero esta clase de información se comprende y se interpreta con más facilidad cuando se muestra en forma de imágenes. La capacidad del ordenador para "volver a dibujar" rápidamente una pintura, para incorporar nueva información, distintas alternativas, etc., y para emitir textos (impresos) cuando sea requerido en ese sentido, resulta asimismo de gran valor para algunas aplicaciones de gestión.

El lienzo del ordenador

¿Cómo crea pinturas el ordenador? Para responder a esta pregunta ocupémonos en primer lugar del "lien-

zo" sobre el que pinta el ordenador. Un microordenador produce imágenes en su pantalla de visualización de datos iluminando o "encendiendo" uno o varios puntos en una de diversas posiciones en la pantalla. Estos puntos están dispuestos en líneas a través de la pantalla y en columnas a lo largo de ella, de manera que la situación de cada punto viene dada por el lugar que ocupe en línea y columna. La iluminación de ciertos puntos, mientras los restantes permanecen apagados, da como resultado imágenes específicas. Esto es válido no sólo para las pantallas monocromáticas, sino también para las visualizaciones a color. En este caso las pinturas se forman al hacer que los puntos adquieran los colores adecuados.

Para visualizar una simple letra o un número, el ordenador se vale de una matriz de puntos rectangular. Esta matriz se conoce como "matriz de puntos". En un microordenador típico ésta consistirá probablemente en un bloque de ocho líneas que contengan ocho puntos cada una.

El número de puntos en la pantalla no es el mismo para todos los ordenadores, pero una cuadrícula bastante típica estaría compuesta por 192 líneas de 256 puntos cada una; o sea, 192 líneas y 256 columnas.

Es evidente que cuantos más puntos haya en la pantalla de visualización de datos del ordenador, mayor será la definición de las imágenes. El grado de definición observable al visualizar gráficos se denomina *resolución*. Se dice que un ordenador que puede visualizar 192 líneas de 256 puntos cada una, tiene una resolución de 256×192 . Cuanto más alta sea la resolución —es decir, cuantos más puntos puedan colocarse

en pantalla—menos "granulosa" será la imagen. Todos los microordenadores poseen una densidad máxima de puntos que pueden visualizarse en sus pantallas para gráficos, pero algunos ejemplares también pueden programarse de modo que utilicen matrices de puntos menos densas. Por ejemplo, un microordenador puede poseer una resolución máxima de 640×256 ; o sea, que le es posible alcanzar una densidad máxima de puntos dada por 256 líneas, cada una de las cuales contiene 640 puntos. Sin embargo, también puede programarse para que utilice sólo 320 de estas columnas, es decir, una resolución de 320×256 , o incluso resoluciones inferiores si así se le requiere. En una máquina con capacidad para este tipo de variación, la resolución debe determinarse al comienzo de la parte del programa que produce los gráficos.

Las líneas y las curvas producidas por un sistema de gráficos por ordenador basado en puntos en realidad no son continuas, como lo serían trazadas con un lápiz. Son senderos de puntos iluminados que están más o menos próximos entre sí según la resolución del sistema. Un sistema de baja resolución sólo será capaz de producir curvas toscas, y una línea recta tampoco será perfectamente recta, ya que los puntos en la pantalla forman líneas rectas sólo en ciertas direcciones (a lo largo de líneas y columnas y a través de las diagonales formadas por la matriz de puntos). Para trazar una línea recta el sistema de gráficos debe iluminar los puntos más próximos al sendero de la línea indicada. El resultado puede ser un efecto "en escalera". También en este caso, cuanto más alta sea la resolución del sistema, menos evidentes resultarán estas "escaleras".

Como los gráficos se exhiben en una pantalla de televisión, para proporcionar una continuidad de visualización la imagen debe ser continuamente "refrescada" o "redibujada", porque, de lo contrario, sólo aparecería por un instante y luego desaparecería. Por este motivo, la imagen debe representarse de alguna manera en el ordenador para que, en caso necesario, éste pueda tomarla como referencia. En realidad la repre-

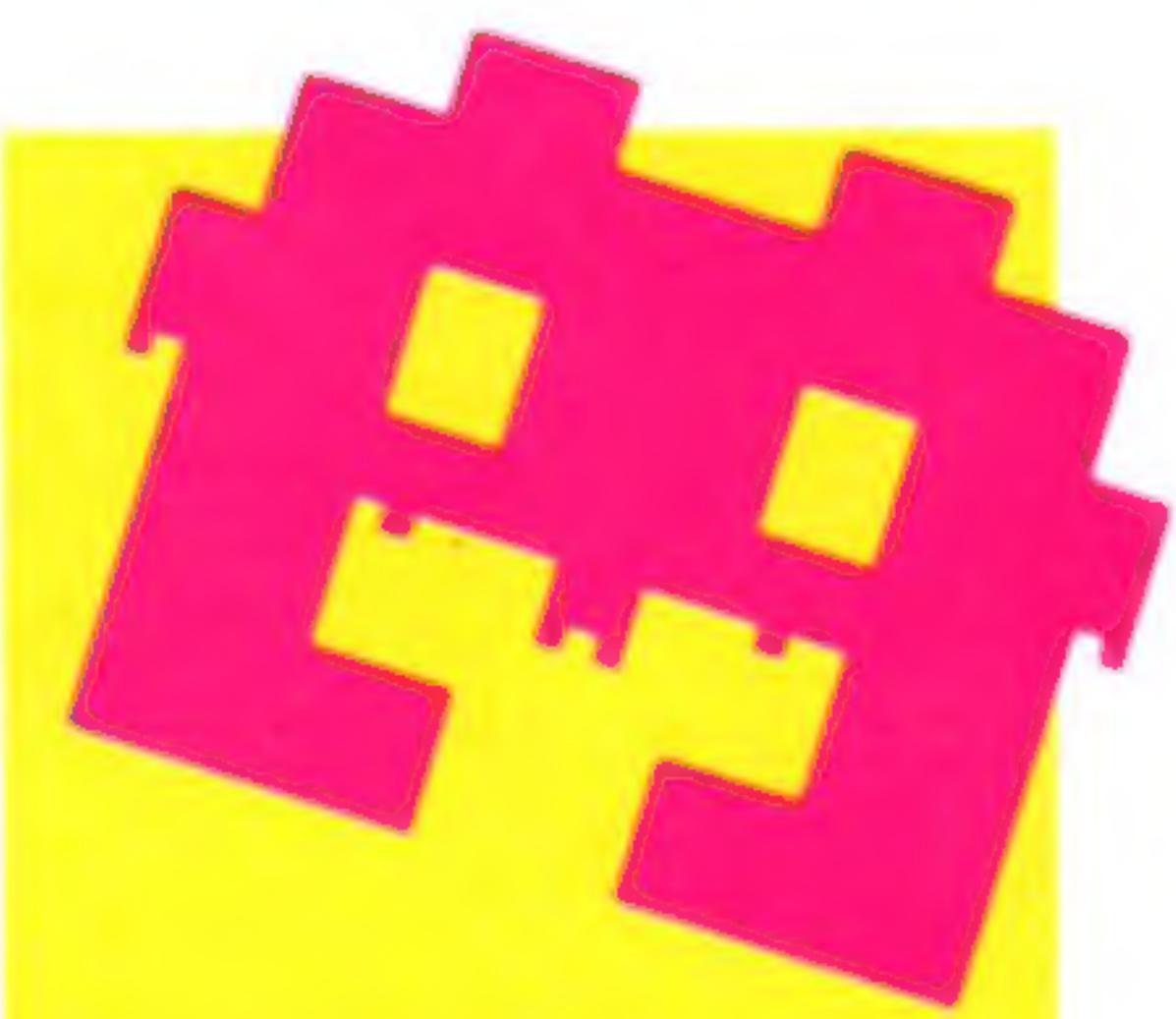
sentación de la imagen se almacena en una zona especial de la memoria del ordenador denominada *memoria de pantalla*. En una visualización monocromática, cada punto de la pantalla corresponde a un bit de la memoria de pantalla. Una imagen se representa estableciendo que los bits que corresponden a los puntos iluminados sean 1 y dejando 0 para los bits correspondientes a los puntos no iluminados. De esta manera, si un ordenador puede mantener una visualización monocromática con una resolución de 256×192 , debe poseer una memoria de pantalla de 256×192 o 49 152 bits, es decir, 6 kilobytes, ya que 1 kilobyte es igual a 8 192 bits.

Para gráficos a color se necesita más memoria. Dos bits pueden representar cuatro colores diferentes, como en el ejemplo siguiente:

bit 1	bit 2	color
0	0	blanco
0	1	rojo
1	0	azul
1	1	negro

Para representar cualquier imagen en cuatro colores, es preciso asignarle dos bits a cada punto de la pantalla. Del mismo modo, para imágenes en ocho colores, a cada punto de la pantalla deben asignársele tres bits, mientras que para visualizaciones en 16 colores corresponderían cuatro bits para cada punto, y así sucesivamente. Por tanto, un microordenador con capacidad para visualizar colores con una resolución de 160×256 , debe tener una memoria de pantalla de $160 \times 256 \times 4$, que equivale a 160 kilobits (20 kilobytes).

La necesidad de la memoria de pantalla explica, en parte, por qué algunos ordenadores están diseñados para funcionar a distintas resoluciones. Si no posee suficiente memoria de pantalla, el ordenador no puede almacenar y, en consecuencia, no puede visualizar imágenes de alta resolución.

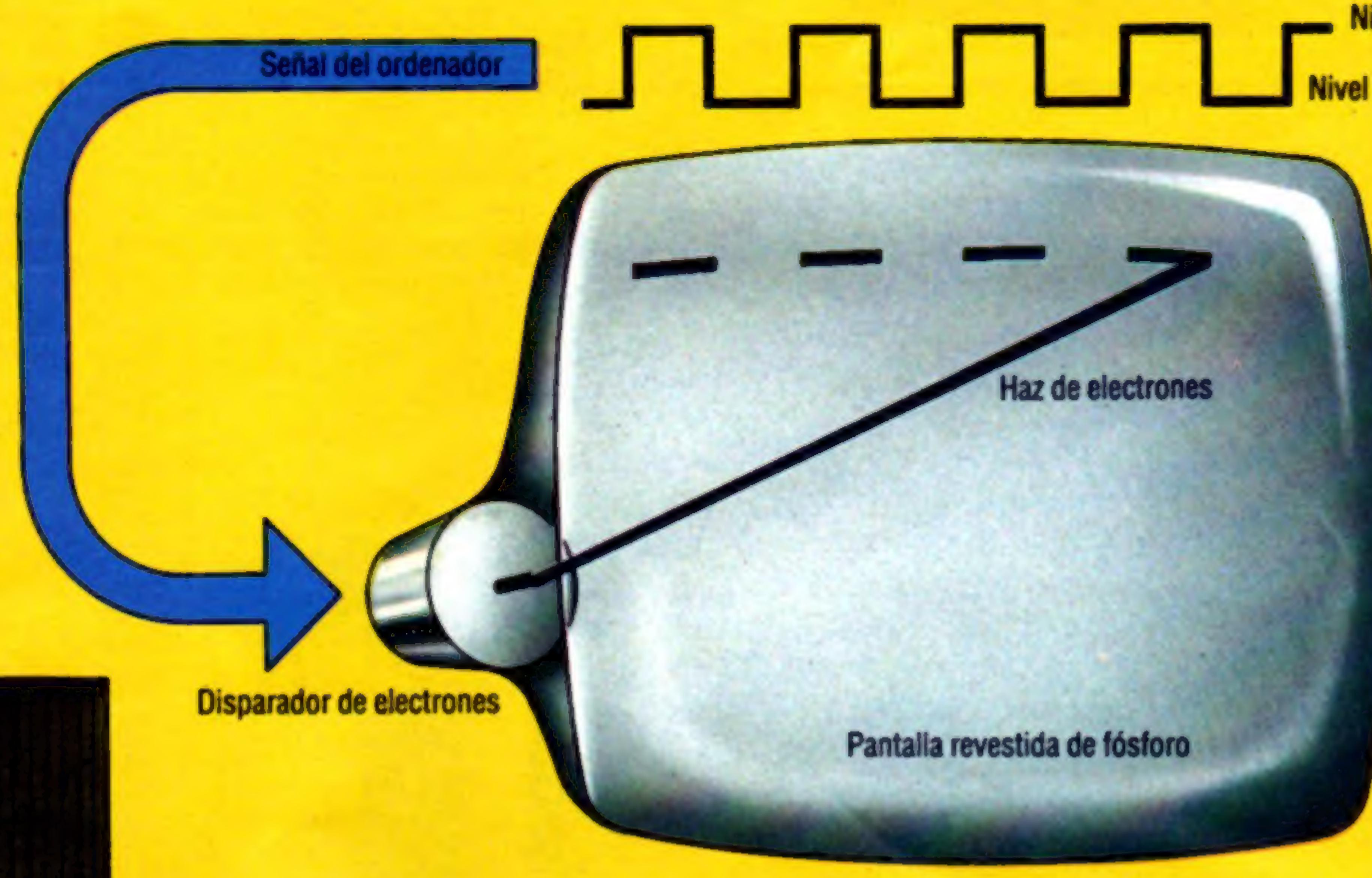


Creando gráficos

Producir imágenes con el ordenador personal puede ser sencillo. La mayoría de los modelos cuentan con órdenes especiales en BASIC para ayudar a "dibujar" o definir en la pantalla tanto a un "invasor extraterrestre" como a un auténtico cuadro. Asimismo, existe un software especial para crear dibujos animados en la pantalla.

Formando imágenes

He aquí cómo las señales producidas por el ordenador se convierten en imágenes en la pantalla del televisor. La imagen de televisión se forma línea a línea, y la señal de entrada que produce la iluminación de las porciones de cada línea está sincronizada para que todas ellas se acoplen para conformar las formas, líneas y caracteres



La señal del ordenador llega al televisor a través del conector de antena. Cuando la señal está arriba, el haz de electrones se enciende mientras explora a través de la pantalla, y se apaga cuando la señal está abajo. El haz se mueve rápidamente a lo ancho de la pantalla, luego salta y comienza ligeramente más abajo, con la línea siguiente. Para cubrir toda la pantalla se requieren cientos de líneas como ésta. Cuando el haz llega hasta la parte inferior de la pantalla, se dirige nuevamente hasta la parte superior y repite el proceso

Bits y bytes

El ordenador sólo comprende números. Pero son números de apariencia un tanto extraña

Siempre que se escribe acerca de ordenadores, se utilizan las palabras *bits* y *bytes*. Estos vocablos describen la manera en que los ordenadores almacenan y emplean los números.

Y lo hacen de un modo bastante diferente al nuestro. Nosotros representamos los números mediante 10 símbolos diferentes (de 0 a 9) y los manejamos en múltiplos de 10. (Esto se conoce como una “base” de 10.) Los ordenadores, por el contrario, para llevar a cabo toda su magia matemática se sirven sólo de dos números: el cero y el uno. Los bits y los bytes son formas de representar combinaciones entre estos dos números.

Un bit es la unidad de información más pequeña que puede manejar un ordenador. Es la forma que tiene el ordenador de representar los números cero y uno. Un grupo de ocho bits se denomina byte; un byte permite al ordenador representar cifras muy elevadas.

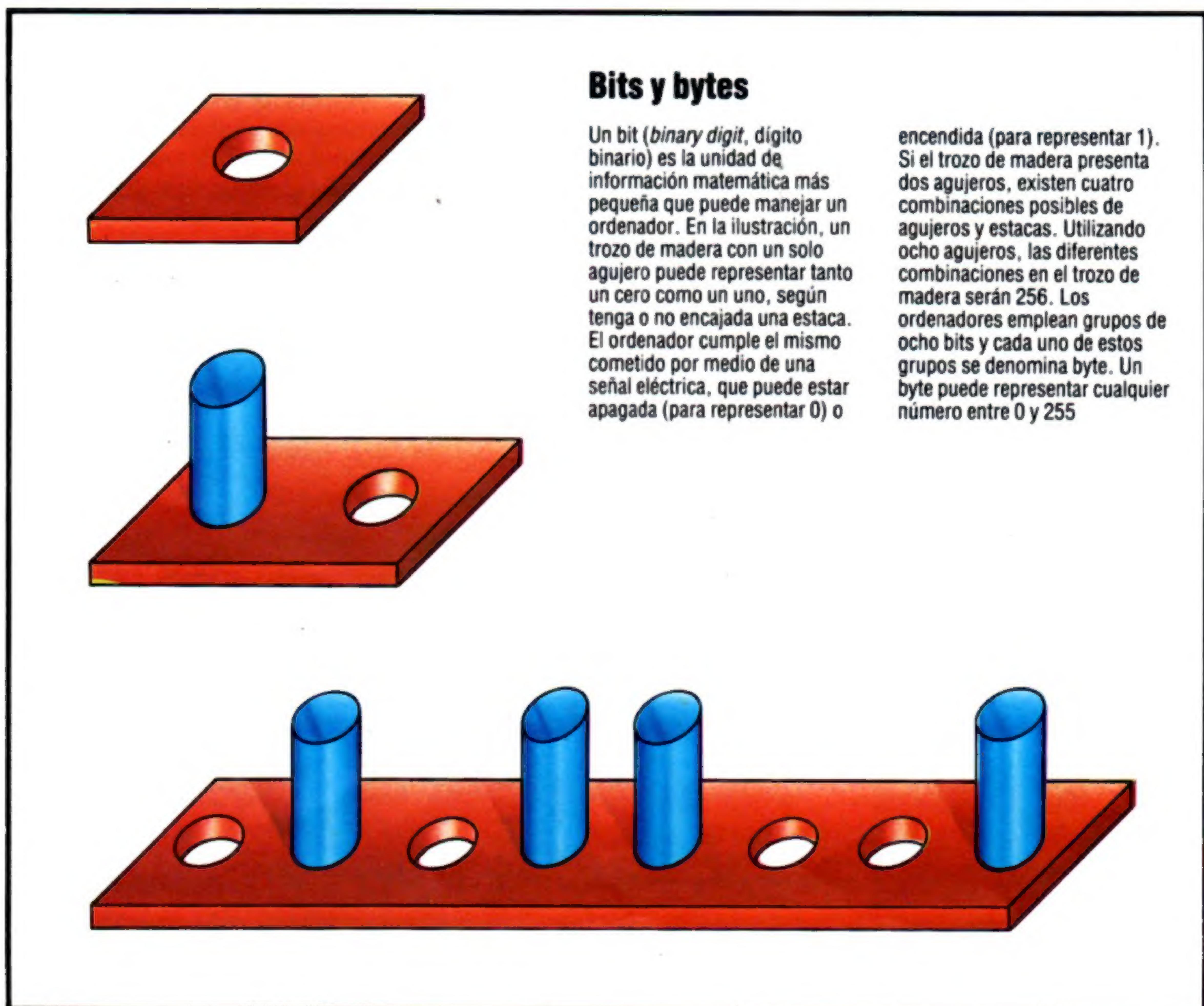
Ocupémonos en primer lugar de los bits: qué son y por qué se denominan así. Los ordenadores son dispositivos electrónicos y, en consecuencia, todas sus funciones las realizan, en última instancia, con señales eléctricas. Una señal eléctrica puede estar “encendi-

da” o “apagada”; éste es el principio por el cual las señales eléctricas pueden representar números.

La ilustración muestra un trozo de madera que presenta un agujero en el cual puede encajarse una estaca. Aun cuando encaje en un solo agujero, la estaca puede representar dos números y constituye una analogía excelente para describir la forma en que funciona un ordenador. El agujero puede no estar ocupado por una estaca, en cuyo caso representa un cero, o bien estar ocupado por ella, en cuyo caso representa un uno. El mismo trozo de madera puede, por tanto, simbolizar tanto un cero como un uno.

En el ordenador se consigue el mismo efecto con una señal eléctrica: si está apagada representa un cero; si está encendida, un uno. Un cable, o un trozo de madera con un agujero, puede, pues, utilizarse para representar dos condiciones: sin estaca o con estaca, ausente o presente, encendido o apagado, 0 o 1.

Esta unidad mínima de información se denomina *bit*. La palabra bit denota su reducido tamaño y representa dos posibles situaciones; deriva de **Bi**nary **digit**. Considerado desde otra perspectiva, un bit



0	0 0 0 0 0 0 0 0	128	1 0 0 0 0 0 0 0
1	0 0 0 0 0 0 1 0	129	1 0 0 0 0 0 1 0
2	0 0 0 0 0 1 0 0	130	1 0 0 0 0 1 0 0
3	0 0 0 0 0 1 1 0	131	1 0 0 0 0 1 1 0
4	0 0 0 0 1 0 0 0	132	1 0 0 0 0 1 0 0
5	0 0 0 0 1 0 1 0	133	1 0 0 0 0 1 0 1
6	0 0 0 0 1 1 0 0	134	1 0 0 0 0 1 1 0
7	0 0 0 0 1 1 1 0	135	1 0 0 0 0 1 1 1
8	0 0 0 1 0 0 0 0	136	1 0 0 0 1 0 0 0
9	0 0 0 1 0 0 1 0	137	1 0 0 0 1 0 0 1
10	0 0 0 1 0 1 0 0	138	1 0 0 0 1 0 1 0
11	0 0 0 1 0 1 1 0	139	1 0 0 0 1 0 1 1
12	0 0 0 1 1 0 0 0	140	1 0 0 0 1 1 0 0
13	0 0 0 1 1 0 1 0	141	1 0 0 0 1 1 0 1
14	0 0 0 1 1 1 0 0	142	1 0 0 0 1 1 1 0
15	0 0 0 1 1 1 1 0	143	1 0 0 0 1 1 1 1
16	0 0 0 1 0 0 0 0	144	1 0 0 1 0 0 0 0
17	0 0 0 1 0 0 0 1	145	1 0 0 1 0 0 0 1
18	0 0 0 1 0 0 1 0	146	1 0 0 1 0 0 1 0
19	0 0 0 1 0 0 1 1	147	1 0 0 1 0 0 1 1
20	0 0 0 1 0 1 0 0	148	1 0 0 1 0 1 0 0
21	0 0 0 1 0 1 0 1	149	1 0 0 1 0 1 0 1
22	0 0 0 1 0 1 1 0	150	1 0 0 1 0 1 1 0
23	0 0 0 1 0 1 1 1	151	1 0 0 1 0 1 1 1
24	0 0 0 1 1 0 0 0	152	1 0 0 1 1 0 0 0
25	0 0 0 1 1 0 0 1	153	1 0 0 1 1 0 0 1
26	0 0 0 1 1 0 1 0	154	1 0 0 1 1 0 1 0
27	0 0 0 1 1 0 1 1	155	1 0 0 1 1 0 1 1
28	0 0 0 1 1 1 0 0	156	1 0 0 1 1 1 0 0
29	0 0 0 1 1 1 0 1	157	1 0 0 1 1 1 0 1
30	0 0 0 1 1 1 1 0	158	1 0 0 1 1 1 1 0
31	0 0 0 1 1 1 1 1	159	1 0 0 1 1 1 1 1
32	0 0 1 0 0 0 0 0	160	1 0 1 0 0 0 0 0
33	0 0 1 0 0 0 0 1	161	1 0 1 0 0 0 0 1
34	0 0 1 0 0 0 1 0	162	1 0 1 0 0 0 1 0
35	0 0 1 0 0 0 1 1	163	1 0 1 0 0 0 1 1
36	0 0 1 0 0 1 0 0	164	1 0 1 0 0 1 0 0
37	0 0 1 0 0 1 0 1	165	1 0 1 0 0 1 0 1
38	0 0 1 0 0 1 1 0	166	1 0 1 0 0 1 1 0
39	0 0 1 0 0 1 1 1	167	1 0 1 0 0 1 1 1
40	0 0 1 0 1 0 0 0	168	1 0 1 0 1 0 0 0
41	0 0 1 0 1 0 0 1	169	1 0 1 0 1 0 0 1
42	0 0 1 0 1 0 1 0	170	1 0 1 0 1 0 1 0
43	0 0 1 0 1 0 1 1	171	1 0 1 0 1 0 1 1
44	0 0 1 0 1 1 0 0	172	1 0 1 0 1 1 0 0
45	0 0 1 0 1 1 0 1	173	1 0 1 0 1 1 0 1
46	0 0 1 0 1 1 1 0	174	1 0 1 0 1 1 1 0
47	0 0 1 0 1 1 1 1	175	1 0 1 0 1 1 1 1
48	0 0 1 1 0 0 0 0	176	1 0 1 1 0 0 0 0
49	0 0 1 1 0 0 0 1	177	1 0 1 1 0 0 0 1
50	0 0 1 1 0 0 1 0	178	1 0 1 1 0 0 1 0
51	0 0 1 1 0 0 1 1	179	1 0 1 1 0 0 1 1
52	0 0 1 1 0 1 0 0	180	1 0 1 1 0 1 0 0
53	0 0 1 1 0 1 0 1	181	1 0 1 1 0 1 0 1
54	0 0 1 1 0 1 1 0	182	1 0 1 1 0 1 1 0
55	0 0 1 1 0 1 1 1	183	1 0 1 1 0 1 1 1
56	0 0 1 1 1 0 0 0	184	1 0 1 1 1 0 0 0
57	0 0 1 1 1 0 0 1	185	1 0 1 1 1 0 0 1
58	0 0 1 1 1 0 1 0	186	1 0 1 1 1 0 1 0
59	0 0 1 1 1 0 1 1	187	1 0 1 1 1 0 1 1
60	0 0 1 1 1 1 0 0	188	1 0 1 1 1 1 0 0
61	0 0 1 1 1 1 0 1	189	1 0 1 1 1 1 0 1
62	0 0 1 1 1 1 1 0	190	1 0 1 1 1 1 1 0
63	0 0 1 1 1 1 1 1	191	1 0 1 1 1 1 1 1
64	0 1 0 0 0 0 0 0	192	1 1 0 0 0 0 0 0
65	0 1 0 0 0 0 0 1	193	1 1 0 0 0 0 0 1
66	0 1 0 0 0 0 1 0	194	1 1 0 0 0 0 1 0
67	0 1 0 0 0 0 1 1	195	1 1 0 0 0 0 1 1
68	0 1 0 0 0 1 0 0	196	1 1 0 0 0 1 0 0
69	0 1 0 0 0 1 0 1	197	1 1 0 0 0 1 0 1
70	0 1 0 0 0 1 1 0	198	1 1 0 0 0 1 1 0
71	0 1 0 0 0 1 1 1	199	1 1 0 0 0 1 1 1
72	0 1 0 0 1 0 0 0	200	1 1 0 0 1 0 0 0
73	0 1 0 0 1 0 0 1	201	1 1 0 0 1 0 0 1
74	0 1 0 0 1 0 1 0	202	1 1 0 0 1 0 1 0
75	0 1 0 0 1 0 1 1	203	1 1 0 0 1 0 1 1
76	0 1 0 0 1 1 0 0	204	1 1 0 0 1 1 0 0
77	0 1 0 0 1 1 0 1	205	1 1 0 0 1 1 0 1
78	0 1 0 0 1 1 1 0	206	1 1 0 0 1 1 1 0
79	0 1 0 0 1 1 1 1	207	1 1 0 0 1 1 1 1
80	0 1 0 1 0 0 0 0	208	1 1 0 1 0 0 0 0
81	0 1 0 1 0 0 0 1	209	1 1 0 1 0 0 0 1
82	0 1 0 1 0 0 1 0	210	1 1 0 1 0 0 1 0
83	0 1 0 1 0 0 1 1	211	1 1 0 1 0 0 1 1
84	0 1 0 1 0 1 0 0	212	1 1 0 1 0 1 0 0
85	0 1 0 1 0 1 0 1	213	1 1 0 1 0 1 0 1
86	0 1 0 1 0 1 1 0	214	1 1 0 1 0 1 1 0
87	0 1 0 1 0 1 1 1	215	1 1 0 1 0 1 1 1
88	0 1 0 1 1 0 0 0	216	1 1 0 1 1 0 0 0
89	0 1 0 1 1 0 0 1	217	1 1 0 1 1 0 0 1
90	0 1 0 1 1 0 1 0	218	1 1 0 1 1 0 1 0
91	0 1 0 1 1 0 1 1	219	1 1 0 1 1 0 1 1
92	0 1 0 1 1 1 0 0	220	1 1 0 1 1 1 0 0
93	0 1 0 1 1 1 0 1	221	1 1 0 1 1 1 0 1
94	0 1 0 1 1 1 1 0	222	1 1 0 1 1 1 1 0
95	0 1 0 1 1 1 1 1	223	1 1 0 1 1 1 1 1
96	0 1 1 0 0 0 0 0	224	1 1 1 0 0 0 0 0
97	0 1 1 0 0 0 0 1	225	1 1 1 0 0 0 0 1
98	0 1 1 0 0 0 1 0	226	1 1 1 0 0 0 1 0
99	0 1 1 0 0 0 1 1	227	1 1 1 0 0 0 1 1
100	0 1 1 0 0 1 0 0	228	1 1 1 0 0 1 0 0
101	0 1 1 0 0 1 0 1	229	1 1 1 0 0 1 0 1
102	0 1 1 0 0 1 1 0	230	1 1 1 0 0 1 1 0
103	0 1 1 0 0 1 1 1	231	1 1 1 0 0 1 1 1
104	0 1 1 0 1 0 0 0	232	1 1 1 0 1 0 0 0
105	0 1 1 0 1 0 0 1	233	1 1 1 0 1 0 0 1
106	0 1 1 0 1 0 1 0	234	1 1 1 0 1 0 1 0
107	0 1 1 0 1 0 1 1	235	1 1 1 0 1 0 1 1
108	0 1 1 0 1 1 0 0	236	1 1 1 0 1 1 0 0
109	0 1 1 0 1 1 0 1	237	1 1 1 0 1 1 0 1
110	0 1 1 0 1 1 1 0	238	1 1 1 0 1 1 1 0
111	0 1 1 0 1 1 1 1	239	1 1 1 0 1 1 1 1
112	0 1 1 1 0 0 0 0	240	1 1 1 1 0 0 0 0
113	0 1 1 1 0 0 0 1	241	1 1 1 1 0 0 0 1
114	0 1 1 1 0 0 1 0	242	1 1 1 1 0 0 1 0
115	0 1 1 1 0 0 1 1	243	1 1 1 1 0 0 1 1
116	0 1 1 1 0 1 0 0	244	1 1 1 1 0 1 0 0
117	0 1 1 1 0 1 0 1	245	1 1 1 1 0 1 0 1
118	0 1 1 1 0 1 1 0	246	1 1 1 1 0 1 1 0
119	0 1 1 1 0 1 1 1	247	1 1 1 1 0 1 1 1
120	0 1 1 1 1 0 0 0	248	1 1 1 1 1 0 0 0
121	0 1 1 1 1 0 0 1	249	1 1 1 1 1 0 0 1
122	0 1 1 1 1 0 1 0	250	1 1 1 1 1 0 1 0
123	0 1 1 1 1 0 1 1	251	1 1 1 1 1 0 1 1
124	0 1 1 1 1 1 0 0	252	1 1 1 1 1 1 0 0
125			

Oric-1

Este modelo puede crear magníficos gráficos en color y una amplia gama de efectos sonoros

El Oric-1, un pequeño ordenador de fabricación británica, compite con el Sinclair Spectrum tanto en precio como en prestaciones. Posee una atractiva carcasa plástica de color gris y su teclado, con un ángulo de inclinación que resulta muy cómodo para digitar, tiene teclas móviles individuales y permite la escritura al tacto.

Existen dos versiones disponibles; la más cara ofrece 48 Kbytes, suficientes para almacenar complejos programas.

El Oric posee las conexiones habituales para televisor, cassette y otras unidades. Se puede conectar con una impresora y dispone de otro conector para encastrar memoria extra, cartuchos de programa y un modem.

El modem es un accesorio particularmente atractivo. Permite que el Oric pueda comunicarse con otros ordenadores a través del teléfono.

El Oric tiene BASIC incorporado y puede, asimismo, funcionar con otros lenguajes. La versión de 48 K se suministra con FORTH y con BASIC.

Utilizando su BASIC incorporado, el Oric puede producir sonido y gráficos en color. También puede visualizar ocho colores y crear y almacenar caracteres de cualquier forma. Mediante las órdenes PAPER (papel) e INK (tinta), es posible cambiar el color de cualquiera de estos caracteres "definidos" y el color del fondo sobre el cual están fijados.

El sonido del Oric es tan impresionante como sus gráficos. Unas órdenes especiales permiten crear una gran variedad de sonidos y música, con notas y acordes que pueden abarcar seis octavas.

El Oric es un micro barato de gran versatilidad, especialmente indicado para los principiantes y para juegos. Su potencial de expansión lo hace especialmente atractivo y su capacidad de comunicarse a través del teléfono representa un reclamo adicional.

Teclado

El teclado del Oric posee 57 teclas móviles. La disposición de las teclas de letras y números responde al trazado QWERTY estándar. Las teclas ESCAPE y CONTROL están a la izquierda, y DELETE y RETURN a la derecha. La línea inferior del teclado contiene las teclas espaciadoras y de control del cursor. Como la disposición de las teclas es similar a la de una máquina de escribir y se accionan individualmente, con un poco de práctica se puede digitar en este teclado a considerable velocidad.



Conexión RGB

Permite conectar al Oric con monitores en color utilizando señales separadas rojas, verdes y azules (RGB: red, green, blue) para una visualización en pantalla de mejor calidad.

Modulador RF

La señal de video producida por el ordenador no puede alimentarse directamente a la conexión de antena de televisión. Este circuito convierte la señal para adecuarla a un televisor normal.

Conexión televisión

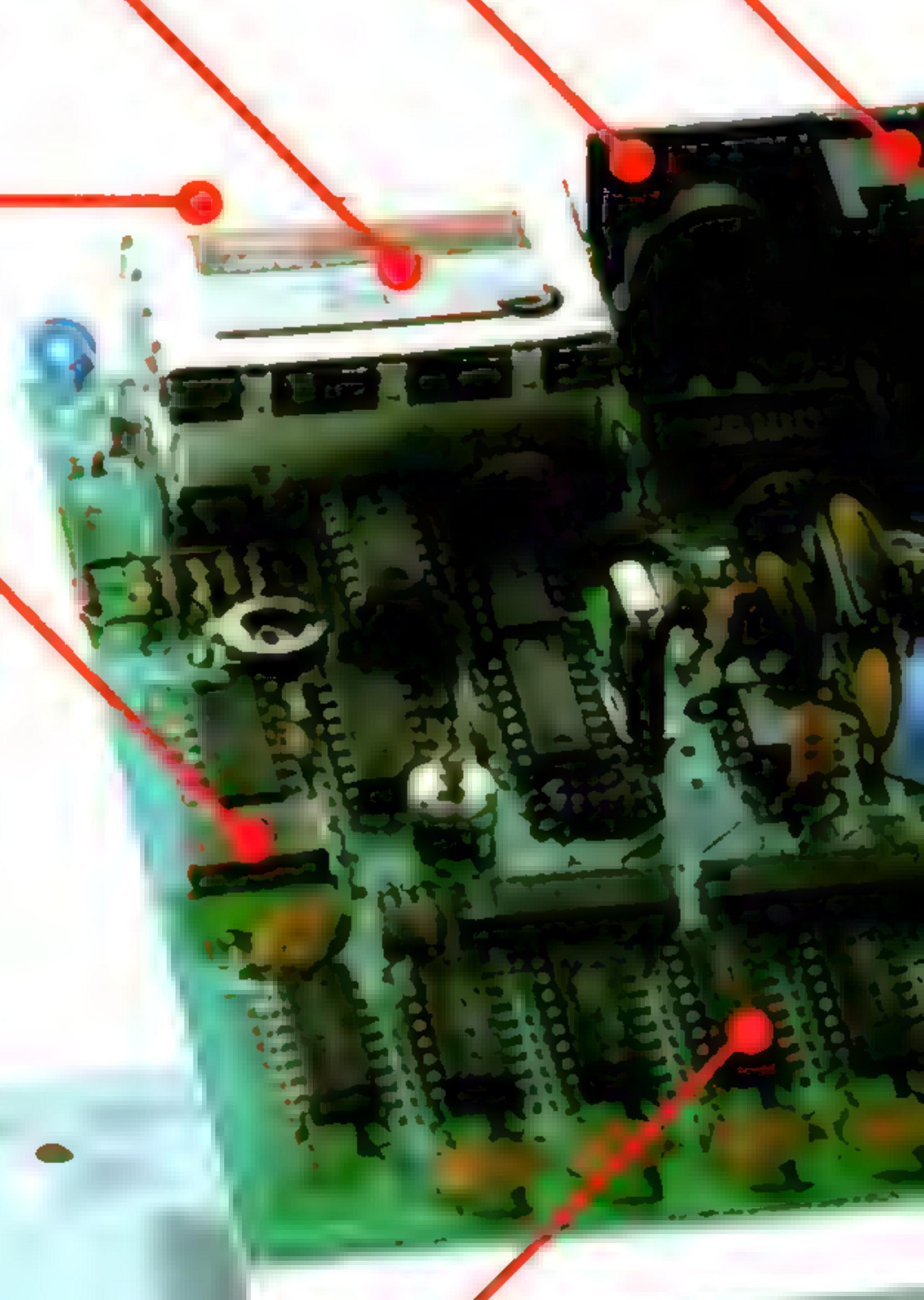
A través de esta conexión el Oric se acopla a un aparato de televisión.

Reloj

El reloj electrónico marca el compás un millón de veces por segundo para regular y sincronizar todas las operaciones que ejecuta el Oric.

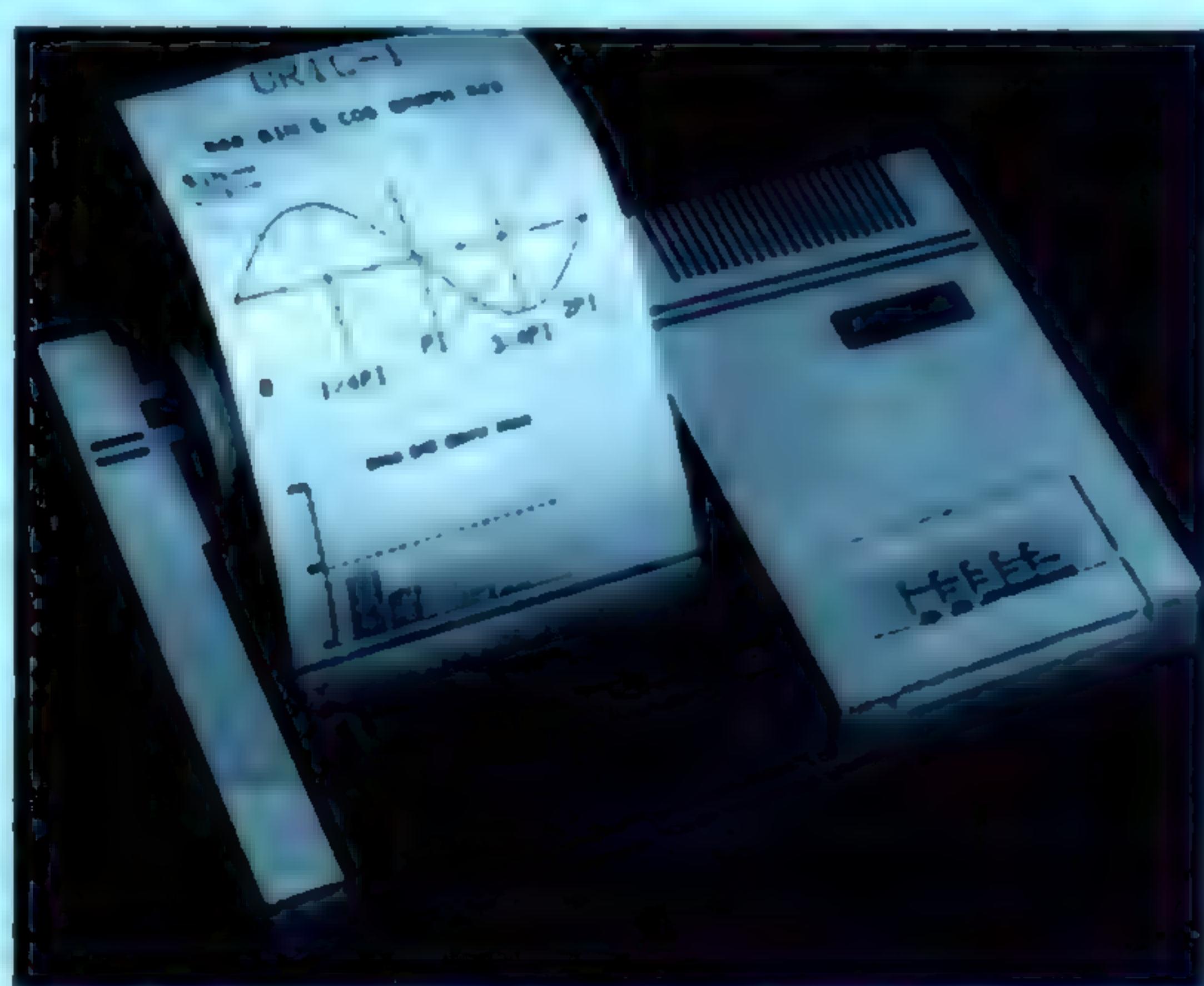
Conexión cassette

La grabadora de cassette se conecta al ordenador por medio de este enchufe.



La impresora

El diseño de la impresora en color armoniza con el Oric-1. Puede imprimir textos y trazar gráficos en cuatro colores: rojo, verde, negro y azul. La impresora utiliza cuatro plumas esferográficas, una para cada color, que escriben sobre un rollo de papel de 11 cm de ancho. Por lo que respecta a la escritura de textos, la realiza con caracteres de 15 tamaños distintos y en cuatro ángulos diferentes. Con este grado de flexibilidad, la impresora, que se conecta mediante interface paralela tipo Centronics Standard, puede efectuar otros muchos cometidos.



RAM

Estos chips constituyen la memoria central del Oric; retienen el software mientras el ordenador está en funcionamiento. La RAM "olvida" su contenido apenas se interrumpe la alimentación eléctrica.





Interface para impresora
Este enchufe proporciona una conexión estándar a través de la cual se puede acoplar al ordenador una impresora

Puerta de expansión
Esta conexión permite adaptar al Oric una gran variedad de dispositivos, incluyendo cartuchos de programas y equipo de comunicaciones

Enchufe red
Aquí se conecta la fuente de alimentación eléctrica proveniente del transformador propio del Oric

Dissipador
Las altas temperaturas pueden dañar con facilidad los circuitos electrónicos. Esta placa disipa todo el calor excesivo que se genera

ROM
La memoria ROM almacena software permanentemente. La función principal de los chips ROM es el almacenamiento del lenguaje de programación BASIC

Micropocesador
Este chip es el caballo de tiro que proporciona al ordenador la capacidad de realizar cálculos. Se trata de un microprocesador 6502A de la Rockwell

Entrada/Salida
Este chip convierte las señales que entran en el Oric de forma que su CPU pueda utilizarlas. La información de salida es convertida en la clase de señal que requieran la impresora y los otros dispositivos

Chip de sonido
Un chip diseñado especialmente para producir notables efectos sonoros

Chip video
El Oric es muy apreciado por su buena resolución en color. El chip genera una señal que contiene información en color. La señal llega hasta la pantalla del televisor a través del modulador

Altavoz
La calidad de sonido es sorprendente. El Oric utiliza su propio altavoz, no el del televisor

ORIC-1

DIMENSIONES 280×178×150 mm

PESO 848 g

CPU 6502A

VELOCIDAD DEL RELOJ 1 MHz

MEMORIA
16 Kbytes de RAM ampliables hasta 48 Kbytes.
16 Kbytes de ROM con BASIC incorporado

VISUALIZACION EN VIDEO
4 modalidades: una modalidad para textos con 28 líneas de 40 caracteres y dos modalidades para gráficos de baja resolución en la misma cuadrícula; modalidad para alta resolución, con 200×240 puntos

INTERFACES
Conexión TV, conexión cassette y alta fidelidad, interface para impresora Centronics, video RGB, conexión de expansión

LENGUAJE SUMINISTRADO
BASIC

OTROS LENGUAJES DISPONIBLES
FORTH

VIENE CON
Unidad de alimentación eléctrica con enchufe integral, conexión TV, conexiones cassette, cassette de juegos, manual

TECLADO
57 teclas móviles individuales, incluyendo barra espaciadora

DOCUMENTACION
Un manual de programación BASIC que describe la máquina y su lenguaje con un estilo muy ágil. Los capítulos relativos a la programación BASIC se refieren a la manipulación de números y palabras y explican cómo utilizar los gráficos en color y los recursos de sonido del Oric para conseguir buenos resultados. Incluye muchos programas cortos que resultan muy instructivos para un primer nivel. Los apéndices indican la manera de organizar la información; no tiene índice. En líneas generales, la documentación es muy adecuada para un principiante

Toma de contacto

Al principio parece que todos los teclados son iguales; sin embargo, unos son mejores que otros y trabajan de forma diferente

El teclado de un ordenador constituye una parte importante del sistema del mismo. Después de todo, es el medio por el que usted puede comunicarse con el ordenador. La importancia del teclado es equiparable a la de la capacidad de memoria, o a la calidad de los gráficos.

En los microordenadores se ha adoptado el teclado tipo QWERTY, similar a los de las máquinas de escribir; este tipo de teclado recibe dicha denominación porque las seis primeras letras de la línea superior forman la palabra QWERTY.

A comienzos de la década de los cincuenta, cuando los ordenadores comenzaron a utilizarse con fines comerciales, el diseño QWERTY, el sistema de mecanografía convencional, se convirtió en el dispositivo de entrada estándar para los ordenadores. En la actualidad el propietario de un ordenador debe someterse al QWERTY, un sistema muy fácil para un mecanógrafo experimentado, pero que para el recién iniciado suele resultar difícil.

Cuando el precio de los ordenadores ascendía a cifras muy altas, el coste de un teclado mecánico era insignificante. Sin embargo, los sucesivos desarrollos en la tecnología de los microprocesadores redujeron notablemente el coste de los componentes electrónicos de los mismos.

Cuando apareció el Sinclair ZX81, un teclado del tipo de una máquina de escribir podía representar una

parte significativa del coste de fabricación de un microordenador. El teclado móvil mecánico, en el que se basa el Dragon, por ejemplo, tiene auténticos interruptores debajo de las teclas (véase la ilustración de la página siguiente). Cuando se pulsa la tecla, los contactos internos se cierran para completar un circuito. Los interruptores de este tipo contienen numerosos componentes y aumentan considerablemente el coste del teclado.

La solución a este problema la constituye un nuevo tipo de teclado, más económico. El teclado "sensible al tacto" del ZX81 se desarrolló a partir de la idea de que la mayoría de las personas que adquieran un microordenador estarían interesadas principalmente en utilizarlo para juegos y para escribir pequeños programas.

Estas aplicaciones implican una actividad mínima con el teclado, por lo cual parecía lógico que los posibles usuarios de un microordenador estuviesen preparados para decidirse por un teclado de inferior calidad. Aunque se prescinda de las ventajas de un teclado convencional del tipo de máquina de escribir, se obtiene, en cambio, un ahorro considerable.

El ZX81 fue diseñado con un teclado sensible al tacto, eliminando la mayoría de las piezas. Este recurso propició el abaratamiento del modelo, pero no representó una solución definitiva. El teclado sensible al tacto ofrece el inconveniente de que no proporciona mucha "realimentación táctil" (o sea, usted nunca está seguro de que la tecla pulsada ha sido registrada en el ordenador, a menos que lo compruebe mirando la pantalla).

En su siguiente producto (el Spectrum), la Sinclair introdujo el teclado de membrana (véase diagrama). Este tipo de teclado representa una nueva mejora, pero todavía carece de la realimentación táctil de que goza el teclado de tipo máquina de escribir.

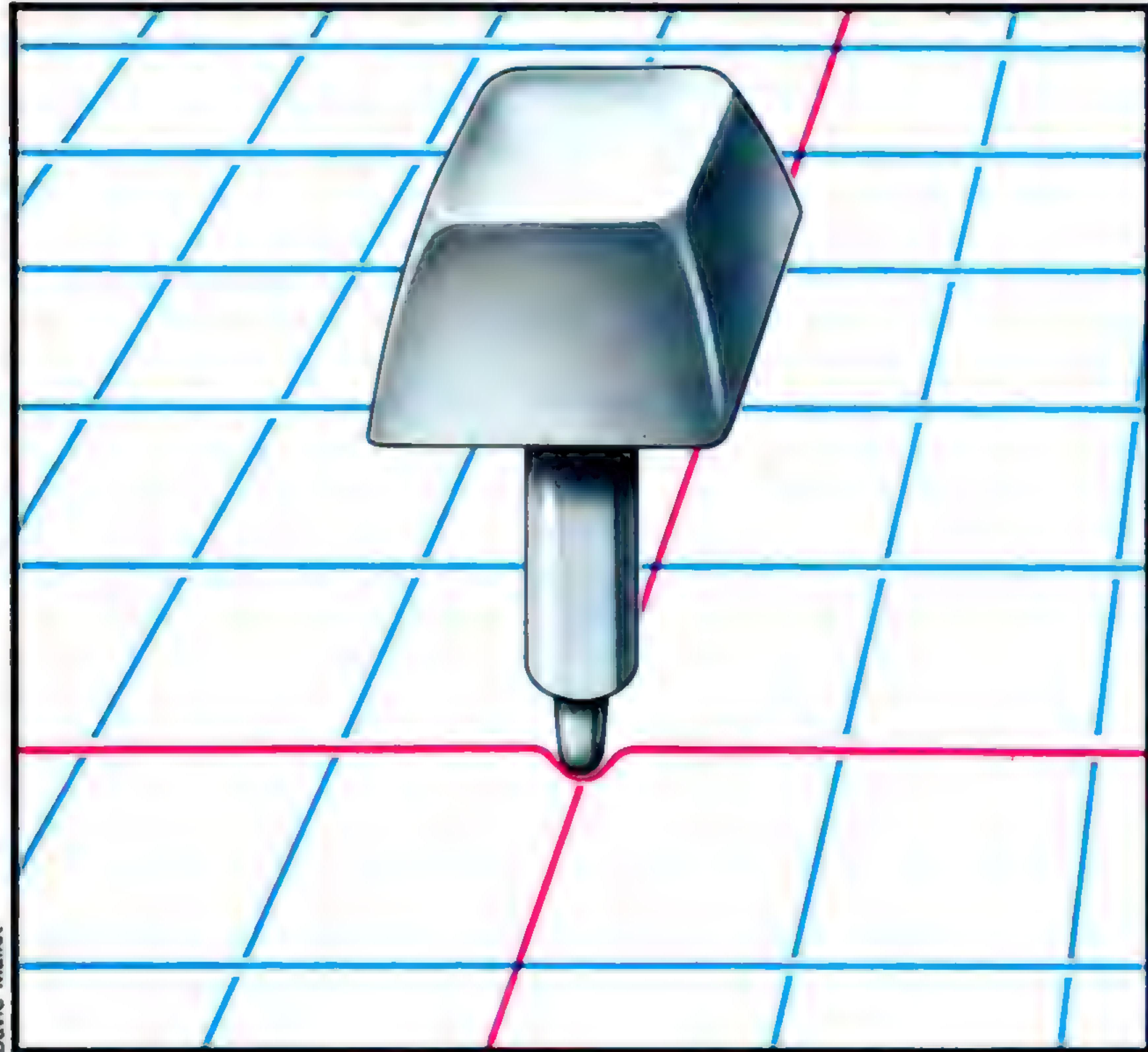
Algunos ordenadores relativamente baratos (entre los que se incluye el Dragon) poseen teclados de máquina de escribir "profesional". Las ventajas de los teclados de este tipo se ponen claramente de manifiesto cuando el ordenador es sometido a una intensa utilización para tratamiento de textos. La familiar sensación de que se está trabajando con una máquina de escribir permite que la tarea se efectúe con muchísima rapidez.

Entre el teclado completamente móvil y el de tipo membrana del Spectrum, existe otro tipo de teclados, que a menudo son denominados "teclados similares al de las máquinas de calcular" (por ejemplo, los que incorpora el New Brain y el Oric-1). Las teclas proporcionan una "sensación" mejor, pero son pequeñas, rígidas y resultan menos aptas para la escritura al tacto que las teclas completamente móviles del tipo de máquina de escribir.

Una forma de superar parcialmente la falta de realimentación táctil de los teclados sensibles al tacto y de

La matriz del teclado

Las teclas de un ordenador son, en realidad, interruptores que están conectados a una rejilla de cables. En la ilustración vemos cómo, cuando es pulsada una tecla, se conectan dos de los cables de la rejilla. A cada tecla corresponde únicamente un par de cables, por lo que sólo efectúa una conexión en la rejilla, permitiendo que el ordenador descubra cuál es la tecla que ha sido pulsada



El teclado Sinclair

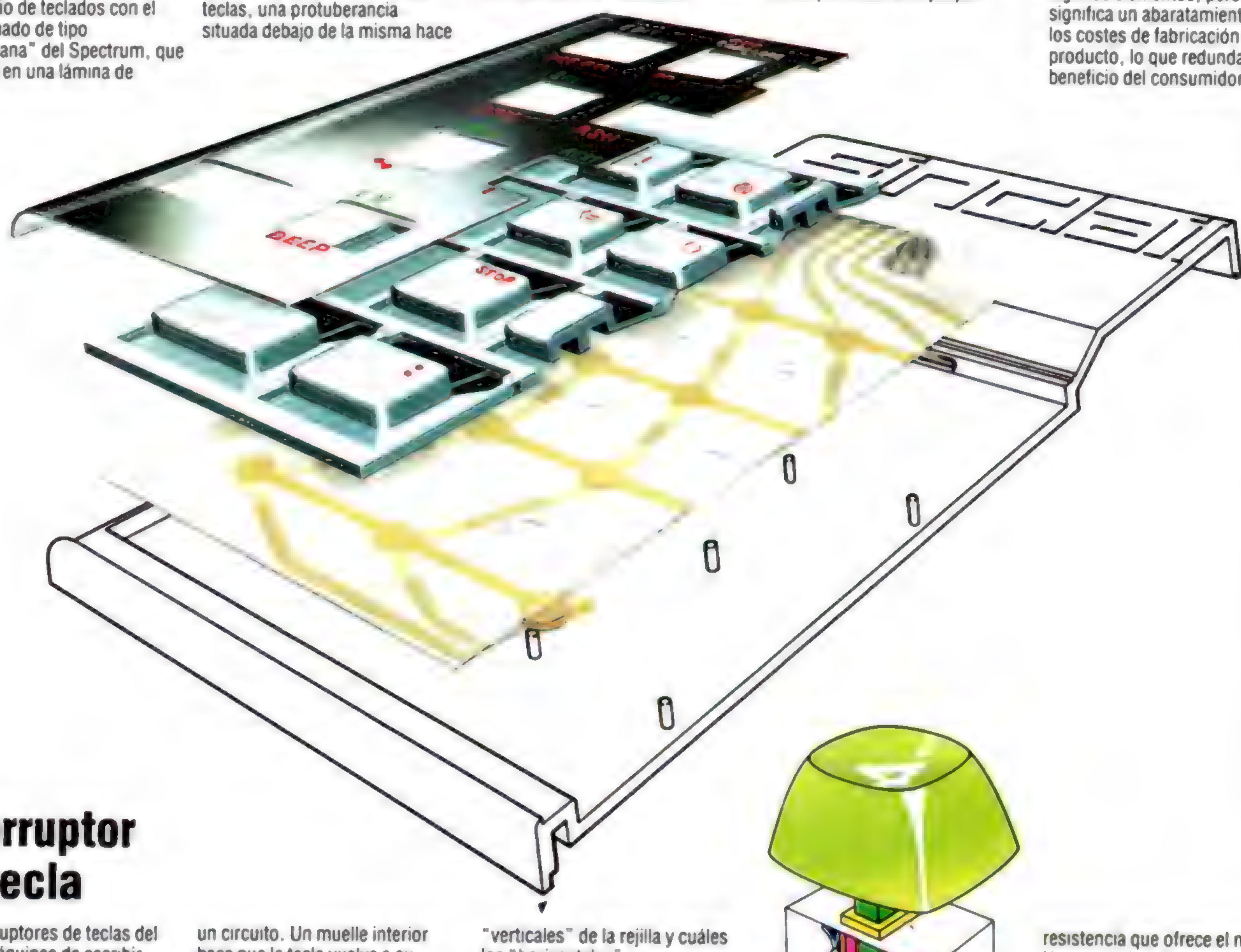
La Sinclair revolucionó el arte del diseño de teclados con el denominado de tipo "membrana" del Spectrum, que consiste en una lámina de

caucho moldeado, con protuberancias en forma de teclas, montada sobre un relleno de contactos que conforman la matriz o la rejilla del teclado. Cuando se pulsa una de estas teclas, una protuberancia situada debajo de la misma hace

que los contactos se unan. El ordenador verifica qué contactos se han cerrado y puede deducir a qué tecla corresponden. Los contactos cerrados por la tecla normalmente se mantienen

separados mediante una burbuja de aire atrapada entre dos láminas de plástico. La fuerza que hace saltar las teclas devolviéndolas a su posición inicial proviene de la propia

elasticidad del caucho, que se estira cuando la tecla es pulsada. Este enfoque original a la ingeniería de teclados ciertamente ha eliminado algunos elementos, pero significa un abaratamiento en los costes de fabricación del producto, lo que redunda en beneficio del consumidor.



Interruptor de tecla

Los interruptores de teclas del tipo de máquinas de escribir incorporan, por lo general, un par de contactos. Estos normalmente están separados y no permiten que pase corriente eléctrica. Cuando se pulsa la tecla, una pieza de plástico moldeado (en color malva en la ilustración) se mueve hacia abajo y permite que los contactos se acerquen y cierren

un circuito. Un muelle interior hace que la tecla vuelva a su posición. Cuando se cierran los contactos se produce un flujo de corriente que es detectado por el ordenador. Los cables conectados a los contactos de cada interruptor están dispuestos en una rejilla. Para saber cuál es la tecla que ha sido pulsada, el ordenador verifica cuáles son los cables

"verticales" de la rejilla y cuáles los "horizontales" que conducen corriente. Las teclas de este tipo son muy complejas desde el punto de vista mecánico y los costes de fabricación son más elevados. Ofrecen una gran fiabilidad y proporcionan una sensación más "positiva" que las teclas de membrana plástica. Esta sensación táctil proviene de la

resistencia que ofrece el muelle. Una tecla bien diseñada proporciona tal realimentación táctil que el usuario sabe si ha pulsado la tecla de forma correcta. Las cabezas de las teclas también están modeladas para que la digitación resulte más cómoda. Los teclados de este tipo son los más indicados si el ordenador ha de utilizarse de forma intensiva.

los del tipo membrana, consiste en emitir un "beep" (sonido corto y agudo) cada vez que se pulsa una tecla. Mediante dicha señal sonora, el usuario puede advertir que la tecla ha sido pulsada y reconocida por el ordenador.

Los diseñadores del Sinclair ZX81 y del Spectrum introdujeron una forma inédita y muy útil de reducir la cantidad de pulsaciones mecanográficas que se precisan para la entrada de programas BASIC. Cada tecla representa algo más que una letra del alfabeto o un simple número. Mediante la utilización de una tecla de "función" especial junto a las teclas normales, puede conseguirse que aparezcan en pantalla palabras completas en BASIC sin necesidad de digitárlas letra por letra. Por ejemplo, la palabra PRINT en BASIC puede ser elaborada simplemente pulsando la tecla de función especial y la tecla de la letra P simultáneamente. En su modelo M5, la firma Sord desarrolla una idea similar.



El educador electrónico

Incluso a los más pequeños de la familia les encantará usar el ordenador. Aquí le explicamos cuál es la mejor forma de iniciarlos

Uno de los instrumentos por ordenador más eficaces para niños a nivel de enseñanza primaria es la "tortuga". El robot está conectado a un microprocesador y funciona según un programa denominado LOGO. Los niños pueden dibujar con la tortuga, que resulta muy útil para enseñar conceptos matemáticos como forma, distancia y la relación entre los objetos. ¡Y además es sumamente divertida!



Ian McKinnell

Muchos padres se preguntan si un ordenador personal podría resultar útil para sus hijos. Gran parte de ellos ya conocen las ventajas que supone para un adolescente aprender el manejo de los ordenadores, tanto en su casa como en la escuela; pero ¿tiene utilidad el uso del ordenador para los niños más pequeños?

¡Sí! La respuesta es decididamente afirmativa, aunque existen diferentes maneras de introducir al niño en el manejo del ordenador, y algunas de ellas mejores que otras.

Muchos de los gobiernos de los países desarrollados, convencidos de la necesidad de que los niños aprendan el manejo de los ordenadores en la escuela, están realizando importantes inversiones para dotar a todos los centros de enseñanza primaria cuando menos de un microordenador. Ahora son los propios maestros quienes deben decidir cuál es la mejor manera de aprovechar este "pequeño potencial" para que resulte útil en sus programas de enseñanza.

Los ordenadores no sólo resultan idóneos para la enseñanza de las matemáticas; con un buen programa, aunque en la actualidad son muy escasos, los ordenadores pueden ayudar al niño a aprender música, ballet, geografía, idiomas y, por supuesto, materias relacionadas con las matemáticas, como aritmética y geometría.

El ordenador puede ayudar a los pequeños de dos formas principales. El niño puede utilizar la máquina para explorar su propio mundo, o bien el ordenador es capaz de actuar como un maestro, instruyendo y adiestrando al niño en una amplia gama de temas educativos.

No constituye una buena idea intentar que su hijo de seis años aprenda a programar un ordenador en BASIC. Antes de los 12 años, un niño no es capaz de comprender los conceptos abstractos de un lenguaje semejante. Pese a que algunos niños están en disposición de escribir programas en BASIC a los nueve años o

incluso antes, los trabajos del psicólogo francés Jean Piaget nos muestran que antes de los 12 o 13 años la mayor parte de los niños tienen dificultades para captar ideas abstractas.

En consideración a este problema, los investigadores han descubierto un sistema para que el niño controle y programe un ordenador sin necesidad de manejar esos conceptos abstractos (véase recuadro relativo al LOGO). La forma habitual en que el maestro introduce al niño en el mundo del ordenador constituye una mezcla de ambos métodos.

El aprendizaje jugando con la tortuga

Incluso los niños más pequeños pueden utilizar el ordenador para aprender. En la ilustración de la página contigua se ve a un niño jugando con la "tortuga", un robot mecánico que está conectado a un microordenador. Estas tortugas resultan caras y están ideadas para su utilización en la escuela, pero su fundamento es muy simple: el ingenio tiene dos ruedas y está provisto de un lápiz. El niño le comunica a la tortuga que avance a través de una hoja de papel y, a medida que se va desplazando, le indica si debe o no trazar una línea (la "huella" de la tortuga). De esta manera el niño dibuja, instruyendo al robot en cómo debe formar esquinas y unir líneas. Como los niños son estimulados para determinar exactamente qué movimientos debe hacer la tortuga al objeto de que resulte un dibujo con formas específicas, llegan a descubrir por sí mismos los elementos que conforman la geometría básica. Este planteamiento de "autoayuda" constituye la base del método LOGO. La teoría consiste en que las lecciones aprendidas "heurísticamente" (mediante pruebas y errores) se comprenden mucho mejor que cuando se muestran ejemplos.

Estas dos escuelas de pensamiento nos señalan dos formas diferentes de utilizar los ordenadores con los niños. De acuerdo con el LOGO, los niños de nueve o diez años comienzan con el manejo de una versión de la tortuga en la pantalla del ordenador, dibujando formas complicadas y enseñando al robot cómo recordar diversos procedimientos. Cuando un niño "instruye" a la tortuga para que realice líneas o figuras, ya sea sobre una hoja de papel o en la pantalla, no hace otra cosa que programar la máquina. El LOGO es un lenguaje mediante el cual el niño programa antes de que haya desarrollado la capacidad de comprensión de los conceptos abstractos necesarios para muchos lenguajes de ordenador. Por lo tanto, a través del "juego con la tortuga" los niños más pequeños se familiarizan con la idea del manejo del ordenador y pueden explorar su propio entorno.

El otro planteamiento se vale de la evidente "pacienza" del ordenador para enseñar al niño mediante ejemplos.

A los niños que tienen dificultad para comprender un tema o una idea se les suele ayudar con programas de "procedimiento correcto y ejercicios", los cuales formulan preguntas al niño y luego, en un marcador, le indican cuántas ha acertado. Muchos de estos programas son enormemente atractivos, ya que están provistos de excelentes gráficos a color e interesantes melodías o efectos sonoros. Estos cuestionarios estimulan al niño a aprender, y el ordenador no se cansa ni abandona mientras el niño insiste con una respuesta equi-

vocada. Esta pacienza del ordenador se ha mostrado muy valiosa para enseñar a los alumnos de comprensión lenta, y los programas de ejercicios en los que, por ejemplo, el niño ha de seleccionar un sustantivo de entre un grupo de palabras, o formar una palabra a partir de una serie de letras, resultan herramientas muy útiles para la educación. Sin embargo, la utilización del ordenador de esta manera nos lleva a sustituir al maestro humano, y de ahí surge una aseveración importantísima: ningún ordenador puede reemplazar al maestro de carne y hueso. El contacto humano constituye el elemento más importante en el proceso de aprendizaje y, si bien el ordenador es el recurso educativo más eficaz que se conoce, no puede sustituir la enseñanza afectuosa impartida por el maestro.

Los ordenadores proporcionan un gran entretenimiento, por lo que resulta una buena idea que se permita a los niños practicar juegos con ellos. A muchos padres les preocupa la posible alienación que pueden producir en el niño juegos como los "invasores del espacio" o Pac-Man, pero aunque dichos juegos son muy divertidos, no existe la más mínima evidencia que sugiera que el atractivo que ejercen sobre los niños rebase los límites de la simple fascinación.

La lógica del LOGO

Mostramos aquí cómo se construyen formas en la pantalla mediante la utilización del lenguaje LOGO.

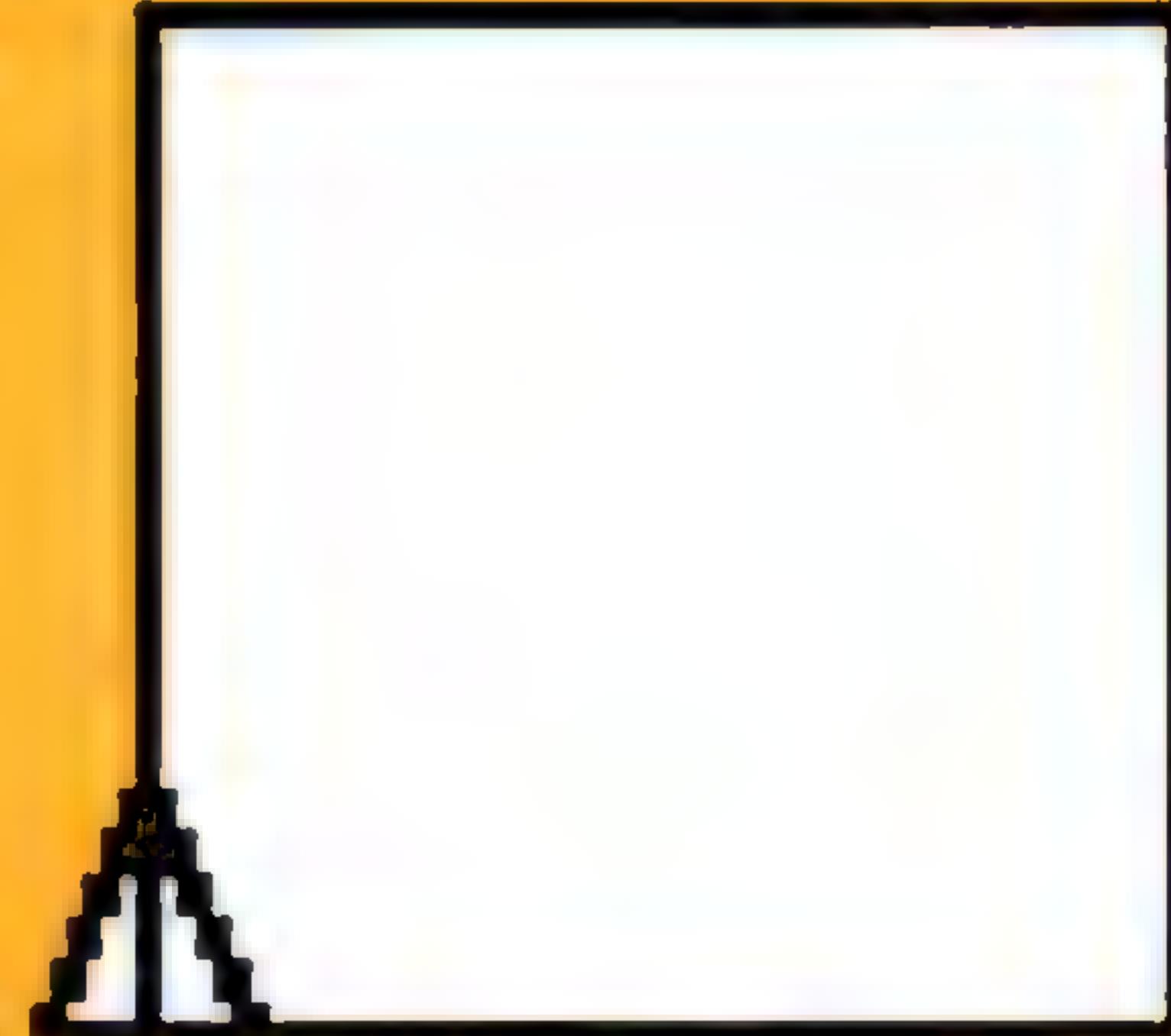
El LOGO es un lenguaje de ordenador desarrollado específicamente para que los niños pequeños (de 4 o 5 años) programen un ordenador. Este lenguaje fue elaborado a finales de la década de los sesenta en el Massachusetts Institute of Technology (Estados Unidos) por un equipo dirigido por Seymour Papert, un matemático que había trabajado con Jean Piaget, el famoso pedagogo, en el centro que éste regentaba en Ginebra.

Para los niños más pequeños, el LOGO se convierte en una "tortuga", ya sea en forma de robot mecánico o como un triángulo luminoso en la pantalla de un ordenador. La orden FORWARD 10 hace que la tortuga se desplace 10 unidades hacia adelante, dibujando una línea. La orden RIGHT 90 determina que la tortuga trace un ángulo recto. Mediante secuencias de órdenes puede conseguirse que la tortuga dibuje cuadrados, triángulos, círculos e incluso formas menos convencionales.

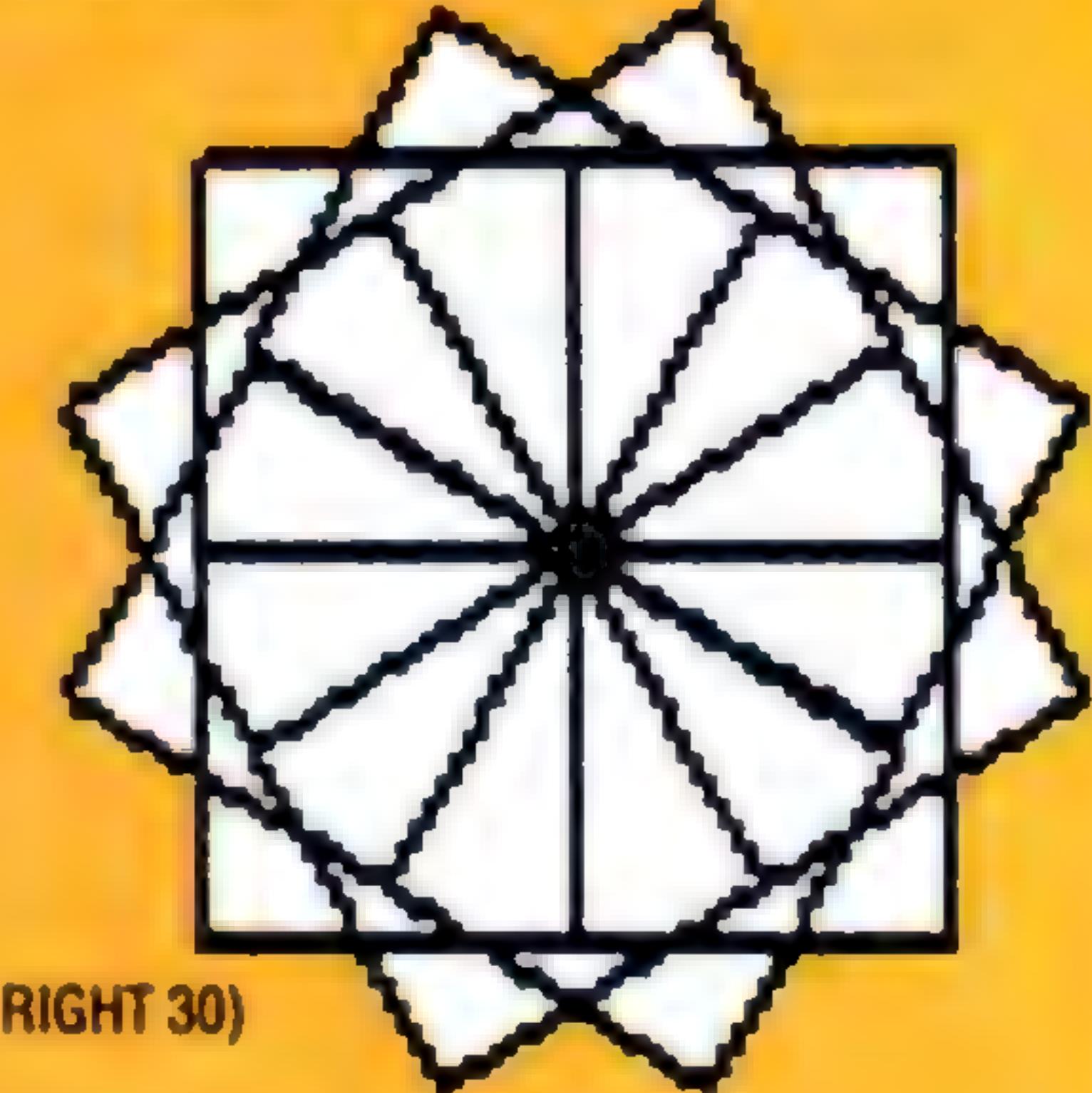
Asimismo, es posible enseñar a la tortuga a "recordar" las órdenes. Los niños, sin advertirlo, mientras enseñan a la tortuga no hacen más que programar un ordenador

Modo de dibujar un cuadrado:
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90

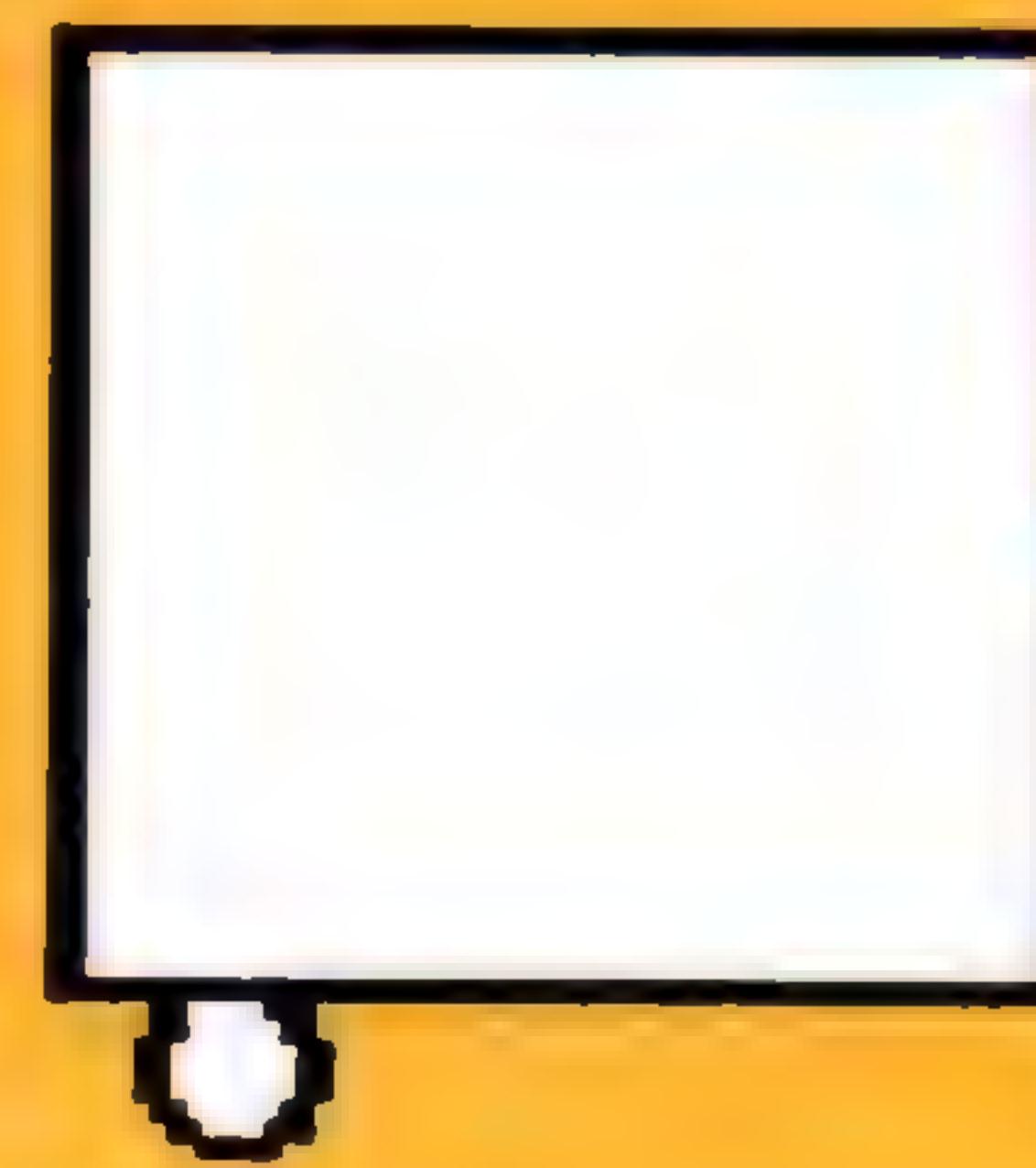
Cómo construir una orden «BOX» (recuadro):
TO BOX
REPEAT 4 (FORWARD 50 RIGHT 90)
END



La orden STAR (estrella):
TO STAR
REPEAT 12 (BOX RIGHT 30)
END



Pueden construirse otras formas muy fácilmente mediante la combinación de la orden BOX con otras similares



Fatiga visual

Algunas personas que trabajan frente a pantallas de ordenador sufren de fatiga visual; pero el período en que los niños permanecen frente a la pantalla no es tan prolongado como para que surja este problema. En el caso de los adultos, el problema parece ser de "acomodación", un fenómeno que se produce cuando el ojo permanece fijo a una distancia focal determinada y precisa de un cierto tiempo para reacomodarse. Si usted advierte que su hijo se concentra demasiado en el ordenador, mirando fijamente a la pantalla durante largos períodos, el problema podrá solucionarse haciéndole que tome un breve descanso cada quince minutos. Es necesario mencionar un problema que plantean los televisores antiguos. Se ha descubierto que algunos televisores en color fabricados antes de 1970 emiten una dosis de baja radiación que puede resultar peligrosa si se trabaja regularmente muy cerca de ellos. Si decide utilizar su antiguo televisor en color para trabajar con el ordenador, es conveniente que compruebe si ha sido fabricado después de 1970.

Los niños menores de siete años necesitan ayuda y supervisión para encender el ordenador y el televisor, y cargar un programa. Si éste es bastante bueno, se puede dejar que lo utilicen por sí solos, aunque esto dependerá de su destreza en la lectura y de su capacidad para responder a las preguntas del programa. Las exigencias en cuanto al hardware son muy sencillas. Un microordenador debe ser sólido: los niños golpean el teclado con los puños cerrados, tiran de las conexiones eléctricas y constantemente tocan la pantalla. Si el sistema es frágil, las conexiones inseguras o resulta difícil de utilizar, no despertará el interés de un niño pequeño. Algunos expertos consideran que el teclado idóneo de un ordenador para niños de corta edad debe ser grande y con teclas claramente definidas. Sin embargo, cuando el niño ya ha desarrollado por completo su capacidad motriz (generalmente a los siete años), es capaz de manipular teclados que incluso resultarían difíciles para un adulto. Los teclados sensibles al tacto de los microordenadores Sinclair más baratos no resultan convenientes para niños menores de 9 o 10 años; por el contrario, las versiones mayores del teclado impreso, como el del Philips Videopac 7000, son adecuadas incluso para niños de cuatro años de edad.

Jóvenes programadores

Para los niños de corta edad la elección del software es más complicada. Si proyecta utilizar un sistema basado en cassettes, usted deberá supervisar el proceso de carga y almacenamiento por completo. Por el contrario, si el sistema que utiliza es a base de discos, comprobará que sus hijos pueden manejar de forma correcta los discos flexibles. Para los más pequeños, niños menores de siete años, uno de los mejores dispositivos para almacenar programas es el cartucho ROM, una caja de plástico que contiene un circuito integrado con un programa fijado eléctricamente. El inconveniente de los sistemas de este tipo consiste en que no permiten al usuario almacenar ningún trabajo; sin embargo, los cartuchos son virtualmente indestructibles, por lo que los niños pueden utilizar el ordenador sin ninguna supervisión.

Si decide adquirir un ordenador exclusivamente para sus hijos, intente colocarlo en un lugar determi-

nado, ya que trasladarlo de una habitación a otra, con la conexión y desconexión de cables que ello supone, si bien no dañará el ordenador (a menos que se le caiga al suelo), posiblemente haga que el niño renuncie a todos esos preparativos y opte por una actividad menos conflictiva, como encender el televisor, por ejemplo.

El centro de trabajo ideal

En una situación ideal, el ordenador del niño debería instalarse en su habitación, junto a su propia pantalla de televisión. Si usted está decidido a que la actitud de sus hijos hacia los ordenadores sea positiva, debe considerar la posibilidad de instalarles un centro de trabajo en alguna de sus habitaciones y adquirir un televisor de segunda mano para su uso exclusivo. (Instale la central de ordenadores en la habitación de su hijo mayor. Es posible que él o ella decidan utilizar el ordenador cuando sus hermanos se hayan ido a dormir.) Un televisor en blanco y negro antiguo puede comprarse por muy poco dinero y, siempre que disponga de sintonizador de canales, resultará adecuado para visualizar en pantalla la información de un ordenador.

Existe una amplia gama de argumentos acerca del efecto del color en los ordenadores destinados a niños pequeños; para algunos especialistas el color constituye un elemento vital, mientras que para otros es sólo un atractivo adicional, aunque innecesario. Resulta obvio que si debe optarse entre una conexión permanente a un televisor en blanco y negro en el dormitorio de los niños, y una conexión temporal al televisor en color de la sala de estar familiar, la instalación de carácter permanente es, indudablemente, la más acertada.

Si usted puede instalar un centro de trabajo/juego con ordenador, ya sea permanente o semipermanente, en una de las habitaciones de sus hijos, constituye una idea acertada disponerlo todo de modo que sea posible trasladar el ordenador sin alterar la instalación.

Cuando proceda a instalar la mesa para el ordenador, asegure los cables y conexiones, de modo que los niños no los arranquen accidentalmente. (Asegúrese asimismo de que todos los enchufes estén bien protegidos y cubiertos por cinta aislante para que no ofrezcan peligro. Y cuide que el televisor no esté conectado a la antena; de lo contrario, puede aparecer "accidentalmente" en la pantalla la última película, cuando los niños se hayan ido a la cama.) Es muy importante que el ordenador rebose en una base estable y no oscile hacia ningún lado. La Sinclair ofrece una bandeja que mantiene en posición fija su ordenador Spectrum, que es muy ligero; si sus hijos son muy traviesos, idee algún sistema, una abrazadera o algo similar, para fijar el microordenador. Por supuesto, el ordenador que usted adquiera para sus hijos también será utilizado por toda la familia, de modo que, en ese caso, es mejor comprar un duplicado de los cables y (de ser necesario) un segundo paquete de alimentación principal. Éstos son relativamente baratos y le permitirán darles las buenas noches con firmeza a sus hijos, desenchufar el ordenador y la grabadora (o la unidad de disco) de su centro de trabajo en el dormitorio (dejando todos los cables enrollados en el mismo lugar), y enchufarlos en su propio televisor utilizando los duplicados de los cables.

El Big Trak

Aunque parece un carro de combate de juguete, en realidad se trata de un eficaz instrumento de aprendizaje. El Big Trak es un juguete programable mediante un ordenador, que permite al niño planificar con toda precisión los movimientos que desea que efectúe el carro. Este pequeño vehículo es capaz de memorizar hasta 16 pasos y se puede programar para que deambule de una habitación a otra de la casa antes de regresar a su base. Mientras el niño se divierte, el ordenador le ayuda a explorar su mundo físico y a elaborar los pasos individuales necesarios en un programa sencillo para ordenador. Pese a su aspecto agresivo, el Big Trak es del agrado tanto de niños como de niñas.



Rizando el rizo

En esta segunda parte del curso nos ocuparemos de cómo interrumpir un bucle, cómo volver a él una determinada cantidad de veces y de la numeración de las líneas

La primera parte de nuestro curso de programación BASIC terminaba con el programa listado abajo. El programa funcionaba bien pero, en virtud del GOTO de la línea 70, efectuaba una y otra vez un bucle hasta el principio y no terminaba nunca. La única manera de salir del bucle era utilizando la tecla BREAK o la tecla RESET.

Ahora vamos a conocer uno de los modos de salir de un bucle como éste, incorporando una "comparación" en el programa. ("Comparación" es la acción de fijar una condición, para luego comprobar si existe.) La forma habitual de hacerlo consiste en probar con un número que en realidad jamás deseariamos utilizar en el programa. El programa nos permitía digitar un número que luego el ordenador imprimía en la pantalla después de sumarle 1. Podemos decidir que nunca deseariamos dar entrada a un número mayor que 999. En este caso hemos de comparar para ver si el número al cual le hemos dado entrada es mayor que 999. Digite el programa y luego agregue:

35 IF A > 999 THEN GOTO 80<CR>

Ahora ponga en marcha otra vez el programa y verá que funciona como antes; a menos, claro está, que dé entrada a un número mayor que 999. Pruebe digitando 1000<CR> y vea qué ocurre.

¿Por qué esta vez el programa se detuvo? La causa está en el IF (si) de la línea 35. Cuando el BASIC encuentra una expresión IF, sabe que a continuación viene una comparación lógica. El signo **>** significa "mayor que". Por lo tanto, la línea 35 significa IF (variable) A (es mayor que) 999 THEN GOTO (línea) 80 ("si la variable A es mayor que 999, luego volver hasta la línea 80"). Si usted digita 1 000, el valor de A se convierte en 1 000, que es mayor que 999, de modo que el programa "luego vuelve hasta" (THEN GOTO) la línea 80, que es el final del programa. Si A no fuera mayor que 999, la parte THEN (luego) de la línea sería ignorada y el programa continuaría con la línea siguiente.

De modo que al hacer funcionar este programa, usted puede dar entrada a todos los números que deseé, con la condición de que no sean mayores que 999. Tan pronto como usted dé entrada a un número mayor que 999, la sentencia IF-THEN lo detecta y concluye el programa yendo hasta END (final). Cuando un programa BASIC llega hasta el final o se termina, se le dará a usted un aviso de "listo" en pantalla. Según sea su ordenador, este aviso tendrá diversas formas. En el Dragon, el aviso es OK. En el Sord, READY. Cualquiera que sea la forma asumida, el aviso de listo es la manera que tiene el BASIC de decirle a usted que no hay ningún programa funcionando y que está a la espera de sus órdenes.

Existen muchas variantes en la forma en que las distintas versiones de BASIC utilizan THEN. Consulte el recuadro de "Complementos al BASIC" en la página 39.

Otras comparaciones que se utilizan en BASIC son < (menor que), = (igual a), >= (mayor o igual a), <= (menor o igual a) y <> (distinto a). A medida que el curso avance, iremos encontrándonos con frecuencia con estas comparaciones.

Antes de seguir adelante, es conveniente realizar algunos ejercicios para acostumbrarse a utilizar estas comparaciones.

Ejercicios

- Cambie una de las líneas de modo que el programa se interrumpa si A = 1000.
- Cambie una de las líneas de modo que el programa se interrumpa si el número al cual le da entrada es menor que cero.
- Cambie la línea GOTO de modo que haga que el programa efectúe un bucle hasta el principio si A es igual a o menor que 500. Una pista: no necesitará una línea IF-THEN y una línea GOTO separadas.

Descubriendo FOR-NEXT

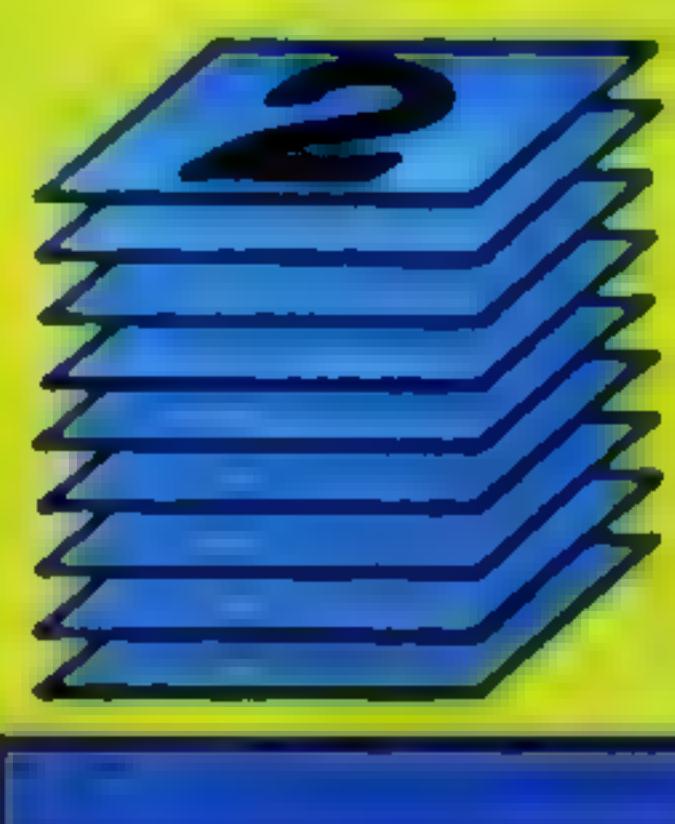
En muchas ocasiones al escribir un programa usted deseará repetir algunos puntos del mismo un determinado número de veces. El GOTO de la línea 70 hacía que el programa efectuara un bucle todas las veces que lo deseáramos. Luego agregamos una sentencia IF-THEN en la línea 35 que nos permitía escaparnos del bucle dando entrada a un número "fuera de escala".

```
10 REM LOS ORDENADORES NUNCA SE EQUIVOCAN
20 PRINT "DIGITE UN NUMERO"
30 INPUT A
40 LET A = A + 1
50 PRINT "CREO QUE EL NUMERO DIGITADO ERA ";
60 PRINT A
70 GOTO 20
80 END
```

El bucle FOR-NEXT en BASIC

En BASIC, la sentencia FOR-NEXT crea un bucle, de modo que esa parte determinada del programa puede repetirse un número exacto de veces. Cuando el BASIC se encuentra con la palabra FOR, sabe que la parte siguiente del programa debe repetirse. Crea una variable (en este caso X) y establece su valor en 1. Se "ejecuta" la parte siguiente del programa (la parte que debe ser repetida). Cuando el BASIC se encuentra con la sentencia NEXT X, en vez de seguir adelante con el resto del programa, realiza un bucle hasta la línea FOR X, aumenta la variable X (representada aquí como la tarjeta siguiente que es depositada en un recipiente para tarjetas denominado X) y luego retoma nuevamente la parte "bucle" del programa. Este proceso se repite 10 veces.

X es el nombre con el que se conoce a la "variable índice". Lleva la cuenta de las veces que el programa ha efectuado un bucle repitiendo la parte que debía repetirse. Tan pronto como el valor de X se ha convertido en 10 (aumenta en 1 cada vez que el programa efectúa un bucle), la siguiente vez que el BASIC llega a NEXT X, hace caso omiso de la instrucción y sigue adelante con el resto del programa.



Esta es la primera parte del programa

FOR X = 1 TO 10

Cuando el BASIC llega a FOR X, "sabe" que a continuación se espera un bucle

PROGRAMA

PROGRAMA

NEXT X

Ésta es la parte del programa que debe repetirse

Vuelve a dirigir el programa hasta FOR X

El programa continúa tras 10 bucles

Sin embargo, tal como aprendimos en la primera parte del curso, en algunas ocasiones utilizar GOTO para hacer un bucle no es adoptar el procedimiento más acertado.

Volvamos a nuestro antiguo programa, ahora modificado para que se ajuste a la verdad, para multiplicar por 10 la entrada del número, y hagámoslo exactamente ocho veces.

```
10 REM MULTIPLICADO POR 10
20 FOR X = 1 TO 8
30 PRINT "DIGITE UN NUMERO"
40 INPUT A
50 LET A = A * 10
60 PRINT "SU NUMERO MULTIPLICADO POR 10 ES: "
70 PRINT A
80 NEXT X
90 END
```

Digite este programa, lístelo (LIST) para verificar que no haya ningún error y luego hágalo funcionar (RUN). Se le solicitará un número sólo ocho veces. Luego el programa se detendrá. La causa de esta detención debemos hallarla en la línea 20.

20 FOR X = 1 TO 8

Esta línea forma parte de un bucle FOR-NEXT. Este bucle constituye una de las más útiles estructuras que ofrece el lenguaje BASIC. Merece que la estudiamos atentamente.

Tal como la hemos utilizado aquí, hemos creado una variable denominada X. (En la primera parte del curso, p. 21, hemos explicado qué es una variable.) Podríamos haberla llamado de cualquier manera (excepto A, que ya la estamos utilizando con otro fin). FOR siempre debe utilizarse seguido de NEXT, pero NEXT aparecerá más tarde en el programa (después del trozo que ha de repetirse). El elemento FOR de un bucle FOR-NEXT tiene siempre la siguiente forma:

FOR variable = valor inicial TO valor final
(para variable = valor inicial hasta valor final)

En nuestro ejemplo FOR X = 1 TO 8 hemos llamado X a la variable y le hemos dado un valor inicial de 1. El ordenador ejecuta entonces la siguiente parte del programa; el número que digitamos es multiplicado por 10 y luego impreso en la pantalla. Después de esto llegamos a NEXT X y el programa hace un bucle volviendo hasta donde está la variable X, es decir, a la línea 20. Después de hacerlo aumenta X en 1, de modo que X adquiere un valor de 2. Luego se vuelve a ejecutar la parte del programa comprendida entre el bucle FOR-NEXT. Al llegar nuevamente hasta NEXT, en la línea 80, el programa vuelve a hacer un bucle y convierte a X en 3.

El programa continúa repitiéndose de esta manera hasta que X se convierte en 8. Entonces el bucle se termina; NEXT X ya no vuelve hasta FOR X y el programa continúa en la línea siguiente.

Otros usos de los bucles FOR-NEXT

Con frecuencia los bucles FOR-NEXT se utilizan para producir dilaciones en el programa. En algunas ocasiones a usted no le interesaría que todo se realice a la máxima velocidad; en estos casos, puede introducir una "demora". Probablemente habrá advertido que en el programa MULTIPLICADO POR 10 las respuestas eran transmitidas de modo tan rápido que parecían instantáneas. Hagamos que el ordenador parezca tomarse un tiempo para pensar antes de dar las respuestas; para lograrlo, insertaremos una demora valiéndonos de FOR-NEXT. Agregue en su programa las líneas que aparecen en color azul.

```

10 REM MULTIPLICADO POR 10
20 FOR X = 1 TO 8
30 PRINT "DIGITE UN NUMERO"
40 INPUT A
50 LET A = A * 10
52 FOR D = 1 TO 1000
54 NEXT D
60 PRINT "SU NUMERO MULTIPLICADO POR 10 ES";
70 PRINT A
80 NEXT X
90 END

```

Hemos agregado otras dos líneas, 52 y 54, en nuestro bucle FOR-NEXT original. Ahora examinémoslas.

```

52 FOR D = 1 TO 1000
54 NEXT D

```

D se establece en 1 y el programa continúa con la línea siguiente. Ésta es la sentencia NEXT correspondiente. En realidad, en el interior del bucle no sucede nada; el programa se limita a efectuar un bucle hasta la línea 52 y aumenta D a 2. Esto ocurre 1 000 veces antes de que el programa continúe con la parte siguiente, que está imprimiendo la respuesta. Los ordenadores son veloces, pero para todo se requiere un tiempo finito, de manera que efectuar 1 000 veces un bucle lleva una cantidad de tiempo considerable. Los ordenadores difieren en cuanto al tiempo que invierten en efectuar un bucle. En el Epson HX-20, este bucle FOR-NEXT tarda 2,9 segundos, mientras que en el Spectrum tarda 4,5 segundos. Experimente cambiando el número que utiliza como límite máximo en la línea 52.

Para hacer que el ordenador se comporte más como un ser humano, agregue estas tres líneas:

```

56 PRINT "AHORA DEJEME VER..."
57 FOR E = 1 TO 1000
58 NEXT E

```

Liste (LIST) el programa y hágalo funcionar (RUN). Ahora tenemos dos demoras cuya función es sólo dilatar el tiempo del programa.

Agregue estas dos líneas:

```

51 REM ESTE BUCLE PIERDE EL TIEMPO
55 REM ESTE PIERDE AUN MAS TIEMPO

```

Ahora liste (LIST) el programa y examínelo con atención. Observe cómo todas las líneas extras que hemos agregado se han situado exactamente en los lugares precisos. Lo que nos lleva al último punto de esta parte del curso: los números de las líneas.

Comenzamos nuestro programa original con la línea 10 y continuamos saltando de 10 en 10 para cada nueva línea, terminando con la línea 90. Podríamos

haber elegido otros números cualesquiera, por ejemplo 1, 2, 3... 9. Pero, de haberlo hecho, ¿cómo habrían podido intercalar las líneas extras? A los programadores siempre se les ocurren ideas nuevas y mejores que introducir, de modo que dejemos lugar para ellas en estos grandes huecos entre los números de las líneas en las versiones "Mark I" de sus programas. Si lo desea, puede comenzar incluso por la línea 100 e ir saltando de 50 en 50 o de 100 en 100.

Algunas versiones de BASIC incluyen una orden muy útil denominada AUTO. El BASIC del Epson HX-20 la tiene. No obstante, el Dragon 32, los ordenadores Sinclair y el Commodore Vic 20 no cuentan con ella. Si su BASIC posee AUTO, puede ahorrarse mucho tiempo al hacer que los números de las líneas se generen automáticamente. Averigüe si su BASIC posee AUTO digitando:

AUTO 100, 10<CR>

Si su BASIC posee AUTO, en su pantalla se verá:

100

La pantalla muestra el número 100 seguido de un espacio y luego el *cursor*. El cursor es una marca (algunas veces una línea, otras un cuadrado) que muestra en la pantalla dónde aparecerá el carácter siguiente. Usted puede comenzar a entrar la primera línea del programa desde la posición del cursor. Cuando usted pulse <CR> la línea siguiente aparecerá automáticamente, comenzando por el número de línea 110. El AUTO, si su ordenador dispone de él, puede utilizarse tanto solo como con uno o dos "argumentos". La palabra *argumento* es un término matemático. En la expresión $2 + 3 = 5$, los argumentos son 2 y 3. La orden AUTO puede utilizarse sola (por ejemplo, AUTO<CR>), o bien con un argumento (por ejemplo, AUTO 100<CR>) o con dos argumentos (por ejemplo, AUTO 300,50). AUTO empleado solo suele hacer que los números de línea comiencen a partir de 10 y sigan saltando de 10 en 10. Si se utiliza sólo un argumento (por ejemplo, AUTO 100<CR>), el primer número será 100 (en este caso) y luego los números irán saltando de 10 en 10. Si se utilizan dos argumentos, la primera cifra especifica el número de línea inicial y la segunda el número según el cual se irán produciendo los aumentos. AUTO 250,50<CR> dará 250 como número inicial, el número siguiente será 300 y así se irá aumentando, de 50 en 50. Incluso en el más sencillo de los micros, es prácticamente imposible que se dé el caso de que usted se quede sin líneas.

En el próximo apartado de este curso nos ocuparemos especialmente de las diversas maneras en que se puede mejorar la presentación visual de un programa en la pantalla y de los distintos modos de imprimir información.

Complementos al BASIC



La mayoría de los microordenadores pueden utilizar esta instrucción tanto en forma de IF A>999 THEN 80, como en forma de IF A>999 GOTO 80. (El Spectrum utiliza IF A>999 THEN GOTO 80)



No disponen de esta orden el Commodore VIC 20, el Dragon 32 y el Sinclair Spectrum

El código de barras

Esas misteriosas barras sobre las tapas de los libros, en las cubiertas de revistas o fascículos y en los productos de los supermercados transmiten su mensaje a un ordenador y ayudan a llevar el control de almacenaje y distribución más eficazmente

Decodificación de barras



COMIENZO

FIN

La ilustración superior muestra un código de barras, que representa el número 72. A simple vista parece sólo una serie de líneas negras de distinto grosor. En este caso cada serie incluye cinco barras, dos de las cuales son anchas. La posición de las dos líneas anchas en cada serie indica el número. Unas barras adicionales señalan el principio y el final de cada unidad de información individual. En esta ilustración, el número 72 está codificado en dos unidades, el dígito 7 y el dígito 2.

Existen muchas formas diferentes de codificar información en un código de barras. Dado que el grosor de la barra es variable, ésta puede representarse numéricamente como un 1 o un 0. Esto nos conduce directamente a la matemática binaria de los ordenadores. El *Universal Product Code* es un código de barras ligeramente diferente que se utiliza en el comercio. Según este código las barras pueden ser de diversos espesores, se requieren menos barras y la información se lee a partir de la anchura de la línea.

La razón de que los códigos de barras se haya extendido tanto en las librerías y en los supermercados se debe a que pueden ser leídos por una máquina. El librero se vale de un "lápiz" sensible a la luz, que ilumina el código de barras y registra la cantidad de luz reflejada. En contraste con el fondo blanco, las barras negras casi no reflejan luz. La luz reflejada se convierte en una señal eléctrica que es amplificada. Cuando se registra luz se traduce en un 1 binario, y si no se registra luz y por lo tanto no se produce señal, en un 0 binario.

Las barras proporcionan secuencias de ceros, y cuanto más anchas son, más ceros contienen. Del mismo modo, el fondo blanco proporciona secuencias de unos. De esta manera, la "varita mágica" alimenta el ordenador con los patrones de dígitos binarios, a partir de los cuales puede determinar la composición de un código de barras.

¿Se ha preguntado en alguna ocasión qué significan esas barras impresas en los productos empaquetados del supermercado, en las tapas de un libro o en las cubiertas de algunas revistas o fascículos? Pues bien, se trata del *código de barras*, un ingenioso recurso que puede ser leído en una fracción de segundo por una "varita mágica" sensible a la luz y que introduce en un ordenador la información relativa a esos productos. Mediante este sistema se puede saber, en cualquier momento, cuál es el stock, la fecha de fabricación o impresión, la fecha de caducidad, etc., de un determinado producto.

Veamos cómo funciona este control en el caso de los libros de bolsillo. Cada libro que se publica en los principales países del mundo posee un International Standard Book Number (ISBN), que consiste en uno o más dígitos en los cuales constan el idioma o la zona geográfica en los que se publica el libro (84 para España y todos los países de lengua castellana; el 0 y el 1 para Gran Bretaña, Estados Unidos y otros países de lengua inglesa; el 2 para Francia; el 3 para Alemania, Austria y la Suiza de lengua alemana, etc.), entre dos y siete dígitos para identificar al editor, y entre uno y seis dígitos para identificar el título del libro y su edición. Esto arroja un total de nueve dígitos, y luego existe un dígito de control (que el ordenador utiliza para verificar si todos estos dígitos le

han sido proporcionados en el orden correcto).

Para la codificación en barras, los libros se numeran de acuerdo al sistema European Article Numbering (EAN; numeración europea de artículos), que utiliza un total de 13 dígitos (la mayoría de los productos de las tiendas de comestibles suelen utilizar un número corto de ocho dígitos). Los tres primeros dígitos corresponden a la "bandera" de la EAN (para los libros es el 978). Luego viene el ISBN y, por último, un dígito de control EAN alternativo.

En España, el ISBN (junto con su propio dígito de control) también se imprime en números en el código de barras; éstos pueden ser leídos tanto por el ojo humano como por un lector de caracteres ópticos (Optical Character Reader).

Los lectores de caracteres ópticos constituyen otro desarrollo interesante de consecuencias mucho más amplias. En la actualidad, existen máquinas que son capaces de leer literalmente la palabra impresa mediante la exploración óptica de la línea. La señal de salida del lector se acopla al ordenador, el cual puede entonces procesar la información de diversas maneras. Las palabras leídas por el explorador pueden, por ejemplo, visualizarse en la pantalla del ordenador, eliminando la agotadora tarea de mecanografiarlas. Y todo ello gracias a este ingenio que está revolucionando el mundo: el ordenador.

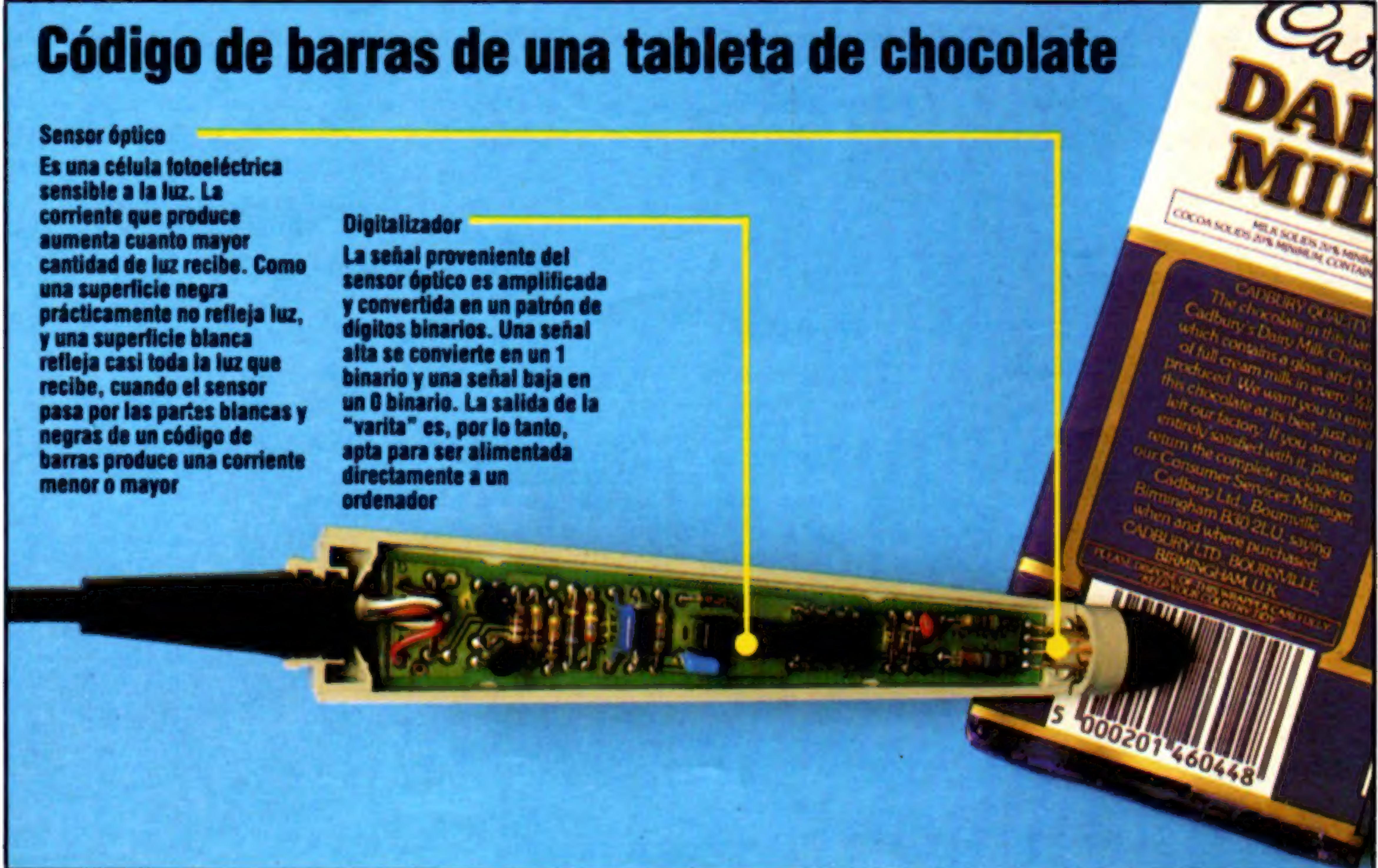
Código de barras de una tableta de chocolate

Sensor óptico

Es una célula fotoeléctrica sensible a la luz. La corriente que produce aumenta cuanto mayor cantidad de luz recibe. Como una superficie negra prácticamente no refleja luz, y una superficie blanca refleja casi toda la luz que recibe, cuando el sensor pasa por las partes blancas y negras de un código de barras produce una corriente menor o mayor

Digitalizador

La señal proveniente del sensor óptico es amplificada y convertida en un patrón de dígitos binarios. Una señal alta se convierte en un 1 binario y una señal baja en un 0 binario. La salida de la "varita" es, por lo tanto, apta para ser alimentada directamente a un ordenador



Ian McKinnell

DONDE CONSEGUIR TU

sinclair

ALAVA

COMPONENTES ELECTRONICOS GAZTEZ
Domingo Beltran, 58 (Vitoria)

DEL CAZ

Avda. Gazteiz, 58 (Vitoria)

VALBUENA

Virgen Blanca, 1 (Vitoria)

ALBACETE

ELECTRO MIGUEL
Tresfrente Gallego, 27

TECON

Maria Marin, 13

ALICANTE

ASEMCA (Villena)
Avda. de la Constitucion, 54 (Villena)

CONSULTING DESARROLLO INFORMATICO

Pais Valencia, 54 (Alcoy)

COMPONENTES ELECTRONICOS LASER

Jaime M. Buch, 7

ELECTRODATA LEVANTE

San Vicente, 28

ELECTRONICA AITANA

Limon, s/n. Edificio Urgull (Benidorm)

ELECTRONICA OHMIO

Avda. El Hamed, 1

LIBRERIA LLORENS

Alameda, 50 (Alcoy)

AVILA

FELIX ALONSO

San Segundo, 15

BADAJOZ

MECANIZACION EXTREMEÑA

Vicente Barantes, 18

SONYTEL

Villanueva, 16

BARCELONA

ARTO

C/ Angli, 43

BERENGUERAS

C/ Diputacion, 219

CATALANA D'ORDINADORS

C/ Trafalgar, 70

CECSA

C/ Mallorca, 367

COMPUTERLAND

C/ Infanta Carlota, 89

COMPUTERLAND

Trav. de Dalt, 4

COPIDUX

C/ Dos de Mayo, 234

D.P. 2000

C/ Sabino de Arana, 22-24

DIOTRONIC

C/ Conde Borrell, 108

EL CORTE INGLES

Avda. Diagonal, 617-619

EL CORTE INGLES

Pza. Cataluña, 14

ELECTRONICA H. S.

C/ S. Jose Oriol, 9

ELECTRONICA SAUQUET

C/ Guilleries, 10

ELEKTROCOMPUTER

Via Augusta, 120

EXPOCOM

C/ Villarroel, 68

GUIBERNAU

C/ Sepulveda, 104

INSTA-DATA

P.º S. Juan, 115

MAGIAL

C/ Sicilia, 253

MANUEL SANCHEZ

Pza. Major, 40 (Vic)

ILLIWATTS

C/ Meléndez, 55 (Mataró)

ONDA RADIO

Gran Via, 581

RADIO ARGANY

C/ Borrell, 45

RADIO SONDA

Avda. Abad Margat, 77 (Tarrasa)

RAMEL ELECTRONICA

Cr. de Vic, 3 (Manresa)

REDIS GESTION

Avda. Sarrià, 52-54

RIFE ELECTRONICA

C/ Aribau, 80, 5.e, 1.a

SERVICIOS ELECTRONICOS VALLES

Pza. del Gas, 7 (Sabadell)

SISTEMA

C/ Balmes, 434

S. E. SOLE

C/ Muntaner, 10

SUMINISTROS VALLPARADIS

C/ Dr. Ferren, 172 (Tarrasa)

TECNOHIFI, S. A.

C/ La Rambla, 19

VIDEOCOMPUT

P.º Pep Ventura, 9, Bl. C. Bjos. Bis (Vic)

BURGOS

COMELECTRIC

Calzada, 7

ELECTROSON

Conde don Sancho, 6

CACERES

ECO CACERES

Diego Maria Crehuet, 10-12

CADIZ

ALMACENES MARISOL

Camoens, 11 (Ceuta)

INFORSA

Avda. Fuerzas Armadas, 1 (Algeciras)

ELECTRONICA VALMAR

Ciudad de Santander, 8

M. R. CONSULTORES

Multi Centro Merca 80 (Jerez de la Frontera)

PEDRO VAREA

Porvera, 36 (Jerez de la Frontera)

LEO COMPUTER

Garcia Escamez, 3

SONYTEL

Queipo de Llano, 17

SONYTEL

Jose Luis Diez, 7

TLC Y AUTOMATICA

Dr. Herrera Quevedo, 2

CASTELLON

NOU DESPACH'S

Rey D. Jaime, 74

Ciudad Real

COMERCIAL R. P.

Travesera de Coso, 2 (Valdepeñas)

ECO CIUDAD REAL

Calatrava, 8

CORDOBA

ANDALUZA DE ELECTRONICA

Felipe II, 15

CONTROL

Conde de Torres Cabrera, 9

ELECTRONICA PADILLA

Sevilla, 9

MORM

Plaza Colón, 13

SONYTEL

Arte, 3

Avda. de los Mozárabes, 7

CUENCA

SONYTEL

Dalmacio Garcia Izcara, 4

GERONA

AUDIFILM

C/ Albareda, 15

CENTRE DE CALCUL DE CATALUNYA

C/ Barcelona, 35

S. E. SOLE

C/ Sta Eugenia, 59

GRANADA

INFORMATICA Y ELECTRONICA

Mejchor Almagro, 8

SONYTEL

Manuel de Falla, 3

TECNIGAR

Ancha de Gracia, 11

GRANOLLERS

COMERCIAL CLAPERA

C/ Maria Maspons, 4

GUIPUZCOA

ANGEL IGLESIAS

Sancho el Sabio, 7-9

BHP NORTE

Ramón M. Lili, 9

ELECTROBON

Reina Regente, 4

HUELVA

SONYTEL

Ruiz de Alda, 3

HUESCA

ELECTRONICA BARREU

M.º Auxiliadora, 1

IBIZA

IBITEC

C/ Aragón, 76

JAEN

CARMELO MILLA

Coca de la Piñera, 3

MARA ILUMINACION

Avda. Linares, 13 (Ubeda)

MICROJISA

Garcia Rebull, 8

SONYTEL



16K: 39.900 Ptas.
48K: 52.000 Ptas.

sinclair

ZX Spectrum

El ordenador de todos para todo.



DISTRIBUIDOR
EXCLUSIVO:

INVESTRONICA

Central Comercial: TOMAS BRETON, 60 - TELF. 468 03 00 - TELEX 23399 IYCO E - MADRID
Delegación Cataluña: MUNTANER, 565 - TELF. 212 68 00 - BARCELONA